Métodos Bootstrap y sus aplicaciones

Javier Belío Miranda

Trabajo de fin de grado en Matemáticas
Universidad de Zaragoza

Directora del trabajo: Ana C. Cebrián Guajardo

Prólogo

El objetivo de la Inferencia Estadística es inferir propiedades generales de una población a partir de lo observado en una muestra. Sin embargo, no solo es estimar el valor del parámetro, que represente la propiedad de interés, sino también cuantificar la imprecisión asociada a esa estimación. La inferencia clásica asume que los datos siguen determinadas hipótesis y modelos de probabilidad para, a partir de ellos, construir estimadores y caracterizar de forma analítica la distribución de probabilidad de dichos estimadores.

Los **métodos bootstrap** introducidos en 1979 por Bradley Efron supusieron un hito fundamental en Estadística ya que establecieron un nuevo marco para el análisis estadístico basado en simulaciones al proporcionar técnicas de inferencia en situaciones más realistas, que requieren menos hipótesis de partida y no requieren obtener la distribución de probabilidad teórica de los estimadores. La idea clave es remuestrear a partir de una muestra, directamente o a través de modelos ajustados, para crear réplicas de conjuntos de datos a partir de los cuales se puede evaluar la variabilidad de los estimadores de interés sin un análisis analítico complejo.

En el **Capítulo 1** se revisan los conceptos básicos del bootstrap; se define la función de distribución empírica $\widehat{F}$ de la población; se muestra que muchos estadísticos se pueden ver como estimadores plug-in, método consistente en estimar algún parámetro función de F, la distribución de la población, evaluando la correspondiente función en $\widehat{F}$; se describe cómo se obtienen las muestras para la aplicación del bootstrap; se revisa la estimación del error estándar para la media muestral; se describe cómo el bootstrap se puede usar para estimar el error estándar y el sesgo de un estadístico; finalmente, se presenta el método jackknife como una aproximación del bootstrap, en el contexto de la estimación de error estándar y del sesgo.

El **Capítulo 2** describe el uso del bootstrap para la construcción de intervalos de confianza. Inicialmente se repasa la teoría clásica a través de la tipificación a una distribución Normal estándar. Posteriormente, se analiza que ventajas tiene el uso del bootstrap en la construcción de intervalos de confianza a través de los intervalos bootstrap-t y de percentiles.

El **Capítulo 3** presenta los contrastes de hipótesis para decidir, basándose en la información proporcionada por la muestra observada, entre dos afirmaciones mutuamente excluyentes. Se comienza introduciendo los elementos básicos de un contraste. Posteriormente, se muestra cómo puede utilizarse el bootstrap en problemas de contrastes de hipótesis simples y compuestas.

El análisis clásico de modelos de regresión y las diferentes aplicaciones del bootstrap en estos modelos se recogen en el **Capítulo 4**. La precisión de los intervalos de confianza de los coeficientes de regresión es analizada a través de simulaciones en el **Capítulo 5**.

Finalmente, el **Apéndice** contiene los ficheros de código R de diferentes programas para los métodos presentados en esta memoria.

Abstract

Statistical theory attempts to answer three basic questions: (1) How should I collect my data? (2) How should I analyse and summarize the data I've collected? (3) How accurate are my data summaries?. Question 3 constitutes part of the process know as statistical inference. The **bootstrap** is a recently developed technique for making certain kinds of statistical inferences introduced in 1979 by Bradly Efron. It is only recently developed because it requires modern computer power to simplify the often intricate calculations of traditional statistical theory.

The particular goal of bootstrap theory is a computer-based implementation of basic statistical concepts. The methods provide inference techniques in more realistic situations, which require fewer initial assumptions and don't require obtaining the theoretical probability distribution of the estimators. The key idea is to resample from the original data, either directly or via a fitted model, to create replicate datasets, from which the variability of the estimators of interest can be assessed without complex analytical calculation.

The use of the term bootstrap derives from the phrase *to pull oneself up by one's bootstrap*, widely thought to be based on one of the eighteenth century Adventures of Baron Munchausen, by Rudolph Erich Raspe. The Baron had fallen to the bottom of a deep lake. Just when it looked like all was lost, he thought to pick himself up by his own bootstraps.

Chapter 1. Introduction to the Bootstrap

Problems of statistical inference often involve estimating some aspects of a probability distribution F on the basis of a random sample drawn from F. The **empirical distribution function**, that we will call $\widehat{F}$, is a simple estimate of the entire distribution F. The obvious way to estimate some interesting aspect of F, like its mean or median, is to use the corresponding aspect of $\widehat{F}$. This is the **plug-in principle** and the bootstrap method is a direct application of this principle. The plug-in principle is a simple method of estimating parameters from samples. The *plug-in estimate* of a parameter $\theta = T(F)$ is defined to be $\hat{\theta} = T(\widehat{F})$.

With classic assumptions, the unknown probability distribution F gives the data $\mathbf{x} = (x_1, ..., x_n)$ by random sampling; from the $\mathbf{x}$ we calculate the statistic of interest $\hat{\theta} = S(\mathbf{x})$. In the bootstrap world, $\widehat{F}$ generates $\mathbf{x}^*$ by random sampling, giving $\hat{\theta}^* = S(\mathbf{x}^*)$. There is only one observed value of $\hat{\theta}$, but we can generate as many bootstrap replications $\hat{\theta}^*$ as affordable. The crucial step in the bootstrap is the process by which we construct from $\mathbf{x}$ an estimate $\widehat{F}$ of the unknown population F.

Summary statistics are often the first outputs of a data analysis. The next thing we want to know is the accuracy of $\hat{\theta}$. The bootstrap has the advantage of being completely automatic and provides accuracy estimates by using the plug-in principle to estimate the **standard error** of a summary statistic no matter how mathematically complicated the estimator may be.

We will estimate the standard error $se_F(\hat{\theta})$ with the standard deviation of the $B \in \mathbb{N}$ replications and obtain that the *ideal bootstrap estimate of the standard error of* $\hat{\theta}$ is

$$\widehat{se}_B(\hat{\theta}) = \left[\frac{1}{B-1} \sum_{b=1}^{B} (\hat{\theta}_b^* - \overline{\hat{\theta}^*})^2 \right]^{1/2} \text{ where } \overline{\hat{\theta}^*} = \frac{1}{B} \sum_{b=1}^{B} \hat{\theta}_b^*.$$

There are other useful measures of statistical accuracy like **bias**, the difference between the expectation of an estimator $\hat{\theta}$ and the quantity θ being estimated. The bootstrap algorithm is easily adapted to give estimates of bias as well as of standard error, obtaining that the *bootstrap estimate of bias* is

$$\hat{\mathfrak{b}}_B(\hat{\theta}) = \overline{\hat{\theta}^*} - T(\widehat{F}).$$

The usual reason to estimate the bias of $\hat{\theta}$ is to correct $\hat{\theta}$ so that it becomes less biased. However, bias correction can be a dangerous practice due to high variability in the estimation of the bias.

The bootstrap has two somewhat different advantages over traditional methods: 1) when used in nonparametric mode, it relieves the analyst from having to make parametric assumptions about the form of the underlying population, and 2) when used in parametric mode, it provides more accurate answers than traditional formulas, and can provide answers in problems for which no traditional formulae exist.

Chapter 2. Confidence intervals

The assessment of uncertainty about parameter values is made using **confidence intervals**. Standard errors are often used to calculate approximate confidence intervals of a parameter θ of interest. Given an estimate $\hat{\theta}$ and an estimate standard error $\widehat{se}$ the *standard confidence interval with confidence level* $100 \cdot (1-2\alpha)\,\%$ is $\left[\hat{\theta} - z_{1-\alpha} \cdot \widehat{se}(\hat{\theta}); \hat{\theta} - z_{\alpha} \cdot \widehat{se}(\hat{\theta})\right]$, where z_α indicates the $100 \cdot \alpha$th percentile point of a standard normal distribution.

Using bootstrap we can obtain accurate intervals without having to make normal assumptions. The **bootstrap-*t* approach** estimates the distribution of $Z = (\hat{\theta} - \theta)/se$ directly from the data, in essence, it builds a table consisting of the percentiles of the B bootstrap samples, and then computes the bootstrap version of Z for each. Finally, the *bootstrap-t confidence interval* is $\left[\hat{\theta} - \hat{t}_{1-\alpha} \cdot \widehat{se}(\hat{\theta}), \hat{\theta} - \hat{t}_{\alpha} \cdot \widehat{se}(\hat{\theta})\right]$, where $\hat{t}_\alpha$ is the estimate of the αth percentile of $Z^* = (\hat{\theta}^* - \hat{\theta})/\widehat{se}$.

The bootstrap-*t* interval is not *transformation-respecting*. It makes a difference which scale is used to construct the interval, and some scales are better than others. For most problems, we don't know which transformation to apply, and this is a major stumbling block to the general use of the bootstrap-*t* for confidence interval construction. In practice bootstrap-*t* can give somewhat erratic results, and can be heavily influenced by a few outlying data points.

There is another approach to bootstrap confidence intervals based on **percentiles** of the bootstrap distribution of a statistic. We have that the $1-2\alpha$ *percentile interval* is $\left[\hat{\theta}^*_{B(\alpha)}, \hat{\theta}^*_{B(1-\alpha)}\right]$, where $\hat{\theta}^*_{B(\alpha)}$ is the $B \cdot \alpha$th value in the ordered list of the B replications of $\hat{\theta}^*$.

The percentile interval for θ agrees well with a standard normal interval constructed on an appropriate transformation of θ and then mapped to the θ scale. The difficulty in improving the standard method in this way is that we need to know a different transformation like logarithm for each parameter θ of interest. The percentile method can be thought of as an algorithm for automatically incorporating such transformations.

Chapter 3. Hypothesis testing

Many statistical applications involve **significance tests** to assess the plausibility of scientific hypotheses. Resampling methods are not new to significance testing, since randomization tests and permutation tests have long been used to provide nonparametric tests. Also Monte Carlo test, which use simulated datasets, are quite commonly used in certain areas of application.

A statistical test is based on a test statistic T which measures the discrepancy between the data and the null hypotheses H_0. If the observed value of the test statistic is denoted by t then the level of evidence against H_0 is measured by the *p-value* $p = \mathbb{P}(T \geq t|H_0)$. However, in most parametric problems and all nonparametric problems, the null hypotheses is composite, that means that it leaves some parameters unknown and therefore does not completely specify F. Therefore the p-value is not generally well defined because it may depend upon which F satisfying H_0 is taken.

There are three solutions to this problem. One is to choose T carefully so that its distribution is the same for all F satisfying H_0. The second and more widely applicable solution is to eliminate the parameters which remain unknown when H_0 is true by conditioning on a sufficient statistic under H_0. The third and less satisfactory approach, which can nevertheless give good approximations, is to estimate F by a cumulative distribution function $\widehat{F}_0$ which satisfies H_0 and calculate $p = \mathbb{P}(T \geq t|\widehat{F}_0)$.

Chapter 4. Regression models

Regression models are among the most useful and most used statistical methods. They allow relatively simple analyses of complicated situations, where we are trying to sort out the effects of many possible *explanatory variables* $X_1, ..., X_p$ in a *response variable* Y.

For the **linear regression model**, the least squares estimation procedure is used to estimate the regression parameters. The model is reasonable and the noise term can be considered to be independent and identically distributed random variables with mean 0, and finite variance σ^2. There are different types of bootstrap methods that work in more general conditions.

There are two basic approaches to bootstrapping in regression. One is **bootstrapping residuals**. For each vector of prediction parameters and response, a residual can be computed. So then it is possible to sample the residuals with replacement, add the bootstrap residuals to the estimates to get a new sample, and then fit the model to the new sample and repeat. This method is easy to do and is often useful. The second is **bootstrapping vectors** that simply treats the vector that includes the response variable and the components of the explanatory variables as though they were independent and identically distributed random vectors.

If the distribution of the estimators of the parameters of the model is not asymptotically normal and we do not know that distribution, bootstrap methods will help us.

Chapter 5. Comparative analysis based on simulations

The objective of this chapter is to analyse the behaviour of the confidence intervals of the estimators of the regression coefficients, obtained with classical and bootstrap methods. In particular, we analyse the confidence level and the length of these confidence intervals under different situations (we will consider different distributions of the model error and different sample sizes). In this way, we want to obtain evidence of which method may be more appropriate in each situation.

If the simulations are done in a manner consistent with the model, the bootstrap will give the same asymptotic results as classical methods. For obtaining interval estimators that are superior to conventional large sample intervals are necessary skewness and kurtosis corrections.

Notación

CONCEPTOS BÁSICOS

- X variable poblacional.
- f función de densidad de la variable X.
- F función de distribución de la variable X.
- $\mathbf{x} = (x_1, ..., x_n)$ muestra aleatoria simple de tamaño n obtenida por muestreo aleatorio de una distribución F.
- $\theta = T(F)$ parámetro, propiedad de la población que nos es de interés. Se enfatiza que el valor θ del parámetro es obtenido mediante un procedimiento de evaluación numérica $T(\cdot)$ a la función de distribución F.
- $\hat{\theta} = S(X)$ estimador del parámetro θ.
- $\mathbb{E}[\theta]$ esperanza de θ.
- $\mathbb{V}ar[\theta]$ varianza de θ.
- $H(\cdot)$ función de Heaviside.
- $\mathbb{1}$ función indicadora o característica.
- $\sim$ denota variable distribuida como o de acuerdo a.
- $\dot{\sim}$ denota variable distribuida aproximadamente a.
- $\overset{i.i.d}{\sim}$ denota muestra de variables aleatorias independientes e idénticamente distribuidas a una distribución.
- $\dot{=}$ denota igualdad aproximadamente a.
- $\#\{A\}$ número de elementos en el conjunto A.

MÉTODO BOOTSTRAP

- $\widehat{F}$ función de distribución empírica.
- $\hat{\theta} = T(\widehat{F})$ estimador del parámetro θ obtenido por el principio plug-in.
- $\mathbf{x}^* = (x_1^*, ..., x_n^*)$ muestra bootstrap obtenida por muestreo aleatorio de una distribución $\widehat{F}$.
- $\hat{\theta}^* = S(\mathbf{x}^*)$ réplica bootstrap del estadístico $\hat{\theta}$.
- B número de réplicas bootstrap.

ERROR ESTÁNDAR DE UN ESTIMADOR

- $se_F(\hat{\theta})$ error estándar del estimador $\hat{\theta}$.
- $\widehat{se}_F(\hat{\theta}) = se_{\widehat{F}}(\hat{\theta})$ estimador bootstrap ideal del error estándar de $\hat{\theta}$.
- $\widehat{se}_B(\hat{\theta})$ estimación bootstrap del error estándar de $\hat{\theta}$ basado en B réplicas bootstrap.

SESGO DE UN ESTIMADOR

- $\mathfrak{b}_F(\hat{\theta})$ sesgo de $\hat{\theta}$.
- $\mathfrak{b}_{\widehat{F}}(\hat{\theta})$ estimador bootstrap ideal del sesgo de $\hat{\theta}$.
- $\hat{\mathfrak{b}}_B(\hat{\theta})$ estimación bootstrap del sesgo de $\hat{\theta}$ basado en B réplicas bootstrap.
- $\overline{\overline{\mathfrak{b}}}_B(\hat{\theta})$ mejor estimador bootstrap del sesgo de $\hat{\theta}$.
- $\hat{\theta}_c$ estimador de θ con sesgo corregido.

MÉTODO JACKKNIFE

- $\mathbf{x}_{(i)} = (x_1, ..., x_{i-1}, x_{i+1}, ..., x_n)$ muestra i-ésima jackknife para $i = 1, ..., n$.
- $\hat{\theta}_{(i)} = S(\mathbf{x}_{(i)})$ réplica i-ésima jackknife del estadístico $\hat{\theta} = S(\mathbf{x})$ para $i = 1, ..., n$.
- $\hat{\theta}_{(\cdot)}$ promedio de las n réplicas jackknife.
- $\widehat{se}_J(\hat{\theta})$ estimador jackknife del error estándar de $\hat{\theta}$.
- $\hat{\mathfrak{b}}_J(\hat{\theta})$ estimador jackknife del sesgo de $\hat{\theta}$.

INTERVALOS DE CONFIANZA

- z_α $100 \cdot \alpha$-ésimo percentil de la distribución $\mathcal{N}(0,1)$, $\alpha \in [0,1]$.
- $t_{n-1;\alpha}$ $100 \cdot \alpha$-ésimo percentil de la distribución t con $n-1$ grados de libertad, $\alpha \in [0,1]$.
- $\widehat{G}$ función de distribución acumulada de $\hat{\theta}^*$.
- $\hat{\theta}^*_\alpha = \widehat{G}^{-1}(\alpha)$ α-ésimo percentil de la distribución de $\hat{\theta}^*$, $\alpha \in [0,1]$.

CONTRASTES DE HIPÓTESIS

- H_0 hipótesis nula.
- H_1 hipótesis alternativa.
- T estadístico test del contraste.
- t valor observado del estadístico test.
- C región crítica o de rechazo.
- α nivel de significación de un test, probabilidad de cometer un error de tipo I.
- p p-valor, nivel de evidencia contra H_0.

MODELOS DE REGRESIÓN

- $X_1, ..., X_p$ variables regresoras o independientes del modelo de regresión.
- Y variable respuesta o dependiente del modelo de regresión.
- $\mathbf{X}$ matriz del diseño del modelo.
- $\boldsymbol{\varepsilon}$ error del modelo.
- Σ matriz de covarianzas.
- $\boldsymbol{\beta} = (\beta_0, ..., \beta_p)^T$ vector de parámetros de regresión.
- $\hat{\boldsymbol{\beta}}$ estimador de mínimos cuadrados de β.
- $\mathbf{e}$ residuo del modelo.
- $\hat{\boldsymbol{\beta}}^*$ estimador de mínimos cuadrados bootstrap de β.

Índice general

Capítulo 1

Introducción al Bootstrap

1.1. Introducción

El objetivo de la Inferencia Estadística es inferir propiedades generales de una población a partir de lo observado en una muestra. De forma más precisa, supongamos que tenemos una muestra y que se quiere estimar un parámetro θ asociado a la distribución de dicha muestra. El objetivo no solo es estimar el valor de ese parámetro sino también cuantificar la imprecisión asociada a esa estimación.

La aproximación tradicional de la inferencia se basa en asumir que los datos siguen determinadas hipótesis y modelos de probabilidad para, a partir de ellos, construir estimadores y caracterizar de forma analítica la distribución de probabilidad (exacta o asintótica) de dichos estimadores. Esta aproximación puede presentar dos problemas, primero que en situaciones complejas, estos cálculos pueden ser complicados o difíciles de obtener. Además, si las hipótesis o simplificaciones planteadas no son adecuadas, los resultados no serán fiables.

Los **métodos bootstrap**, también llamados métodos de remuestreo o computacionalmente intensivos, proporcionan técnicas de inferencia en situaciones más realistas, que requieren menos hipótesis de partida y no requieren obtener la distribución de probabilidad teórica de los estimadores. La idea clave es remuestrear a partir de una muestra -ya sea de forma directa o en un modelo ajustado- para crear réplicas de conjuntos de datos a partir de los cuales se puede evaluar la variabilidad de los estimadores de interés sin un análisis analítico complejo.

El bootstrap fue introducido en 1979 por Bradley Efron y experimentó avances en los siguientes años con aportaciones de otros autores como Robert Tibshirani, Anthony Davison y David Hinkley. El bootstrap es un hito fundamental en Estadística ya que establece un nuevo marco para el análisis estadístico basado en simulaciones.

El origen del término proviene de la expresión inglesa *pull oneself up by one's bootstrap*. Está atribuida a la novela *Relato que hace el Barón de Munchausen de sus campañas y viajes maravillosos por Rusia* de Rudolph Erich Raspe donde el barón cae al fondo de un profundo lago y consigue salir, cuando parece que todo está perdido, tirando de sus propios cordones de las botas (*bootstraps*). El término refleja la propia autosuficiencia del método en el que, en ausencia de otra información de la distribución, la muestra observada contiene toda la información disponible.

En este capítulo se van a introducir técnicas que son aplicables a una única muestra homogénea de datos. Sea una población finita de N individuos con la misma probabilidad de ser escogidos. Consideramos una muestra aleatoria simple $\mathbf{x} = (x_1, ..., x_n)$ de una variable X cuya *función de densidad* y *función de distribución* denotaremos por f y F respectivamente. La muestra se usará para hacer inferencia sobre una propiedad o parámetro de la población, indicado genéricamente por θ, usando un estadístico S cuyo valor en la muestra es s.

En primer lugar, en la siguiente sección se presentan la distribución empírica y el principio plug-in, elementos básicos en este tipo de técnicas. En segundo lugar, se mostrará el principio fundamental del bootstrap y las diferentes aproximaciones que se pueden utilizar para obtener muestras bootstrap. A continuación, se desarrollará la aplicación de estos métodos para el cálculo del error estándar y el sesgo de un estimador. Finalmente introducimos el método jackknife, predecesor del bootstrap, comparándolo con este método.

1.2. La distribución empírica y el principio plug-in

Los problemas de inferencia estadística a menudo implican estimar algunos aspectos de la distribución de probabilidad F a partir de una muestra aleatoria con esa distribución. La **función de distribución empírica**, que denotaremos por $\widehat{F}$, es una estimación de la distribución F. Una de las formas de estimar algún parámetro función de F será evaluar la correspondiente función en $\widehat{F}$. Este método de sustitución es conocido como **principio plug-in** y el bootstrap es aplicación directa de este principio. Usaremos el símbolo $\hat{}$ para indicar cantidades que han sido calculadas a partir de la muestra observada.

1.2.1. La distribución empírica

La *función de distribución empírica* es definida como la distribución discreta que asigna igual probabilidad $1/n$ a cada valor x_i de la muestra para $i = 1,...,n$. Es decir, $\widehat{F}$ asigna a un conjunto A del espacio muestral su probabilidad empírica $\widehat{\mathbb{P}}(A) = \#\{x_i \in A\}/n$, la proporción de la muestra observada $\mathbf{x}$ que ocurre en A. Notar que podemos considerar la función de distribución empírica como la función escalón con un salto de tamaño $1/n$ en cada punto x_i reescribiéndola de las dos siguientes formas:

$$\widehat{F}(z) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}\{x_i \leq z\} = \frac{1}{n}\sum_{i=1}^{n} H(z - x_i),$$

donde $\mathbb{1}$ es la *función indicadora o característica* del suceso $\{x_i \leq z\}$ y $H(u)$ es la *función escalón de Heaviside* que salta de 0 a 1 cuando $u = 0$.

La distribución empírica viene determinada por la lista de valores tomados de la muestra junto con la proporción de veces que ocurre cada valor. En las situaciones en las que se repitan los valores de la muestra, podemos expresar $\widehat{F}$ como el vector de *frecuencias relativas observadas* $\hat{f}_k = \dfrac{\#\{x_i = k\}}{n}$ donde $i = 1,...,n$ y k toma el valor de los diferentes valores de la muestra.

Propiedades del estimador $\widehat{F}$

- **Suficiencia.** Efron y Tibshirani afirman en [12, sección 4.2] que no hay pérdida de información al pasar de la muestra completa al vector de frecuencias ya que $\widehat{F}$ es el estimador suficiente de F, lo que corrobora el hecho de estimar F a través de $\widehat{F}$ cuando no hay otra información de F disponible pues por definición de suficiencia ningún otro estadístico que puede ser calculado sobre la misma muestra proporciona información adicional sobre su valor.

- **Insesgadez y varianza.** Si consideramos la función de distribución empírica como la función escalón y fijamos el valor z, entonces la variable aleatoria $\mathbb{1}\{x_i \leq z\}$ es una variable Bernoulli de parámetro $p = F(z)$. Por tanto, al ser suma de n variables Bernoulli independientes, $n \cdot \widehat{F}(z)$ será una variable aleatoria binomial de media $nF(z)$ y varianza $nF(z)(1 - F(z))$, por lo que $n \cdot \widehat{F}(z) \sim Bi(n, F(z))$. En consecuencia, $\widehat{F}(z)$ es un estimador insesgado de $F(z)$, $\mathbb{E}[\widehat{F}(z)] = F(z)$, con varianza $F(z)(1 - F(z))/n$.

- **Convergencia casi segura.** La Ley Fuerte de Grandes Números de Kolmogorov afirma que dadas $X_1,...,X_n$ v.a.i.i.d. tales que $\mathbb{E}[X_i]=\mu$, entonces $\overline{X}_n \xrightarrow{c.s.} \mu$. Por tanto, tenemos convergencia casi segura de $\widehat{F}$: para todo valor de z, $\widehat{F}(z) \xrightarrow{c.s.} F(z)$.
- **Consistencia.** Entre las propiedades de convergencia casi segura, encontramos que esta convergencia implica convergencia en probabilidad. Por tanto, por definición de consistencia, $\widehat{F}$ es un estimador consistente de F.
- **Convergencia uniforme.** Por el Teorema de Glivenko-Cantelli tenemos el siguiente resultado: $\|\widehat{F}-F\|_\infty = \sup_{z\in\mathbb{R}} |\widehat{F}(z)-F(z)| \xrightarrow[n\to\infty]{c.s.} 0$. Este es un resultado más fuerte que, equivalentemente, nos asegura que la función $\widehat{F}(z)$ converge a $F(z)$ uniformemente sobre z.

1.2.2. El principio plug-in

Las discusiones de inferencia estadística se expresan en términos de parámetros y estadísticos. Un parámetro se puede expresar como una función de la distribución de probabilidad F y un estadístico como una función de X. Es decir, denotaremos $\theta = T(F)$ donde esta notación enfatiza que el valor θ del parámetro es obtenido mediante un procedimiento de evaluación numérica $T(\cdot)$ a la función de distribución F. Para estimar θ, buscaremos un estadístico $S = S(X)$ que llamaremos estimador y cuyo valor en $\mathbf{x}$, $S(\mathbf{x}) = \hat{\theta}$, será la estimación de θ basada en la información disponible.

El principio plug-in es un método de estimación de parámetros y definiremos el *estimador plug-in* de un parámetro $\theta = T(F)$ como $\hat{\theta} = T(\widehat{F})$. En otras palabras, estimaremos la función $\theta = T(F)$ dependiente de la función de distribución F por la misma función evaluada en la distribución empírica $\widehat{F}$. Por lo general el principio plug-in funciona, si la única información disponible sobre F proviene de la muestra $\mathbf{x}$. Bajo esta circunstancia $\hat{\theta} = T(\widehat{F})$ no se podrá mejorar como estimador de $\theta = T(F)$, al menos no en el sentido asintótico usual $n \longrightarrow \infty$.

Como ejemplo sencillo, veamos que la media muestral es un estimador plug-in de la media poblacional. En este caso, $\mu = T(F) = \int y\, dF(y)$, y queremos ver si $\hat{\mu} = T(\widehat{F}) = \overline{Y}_n$. En efecto,

$$T(\widehat{F}) = \int y\, d\widehat{F}(y) = \int y\, d\left(\frac{1}{n}\sum_{i=1}^{n} H(y-y_i)\right) = \frac{1}{n}\sum_{i=1}^{n}\int y\, dH(y-y_i) = \frac{1}{n}\sum_{i=1}^{n} y_i = \overline{y},$$

ya que $\int a(y)\, dH(y-x) = a(x)$ para cualquier función continua $a(\cdot)$.

1.3. El principio básico del bootstrap

La idea básica del bootstrap es crear réplicas de nuestra muestra, que se pueden obtener mediante distintos procedimientos, a partir de las cuales se puede caracterizar la distribución del estimador de interés utilizando su distribución empírica y evitando así complicados y restrictivos cálculos analíticos.

Efron y Tibshirani en [12, sección 8.2] resumen la relación entre la muestra y el bootstrap en el siguiente esquema de dos mundos paralelos que detallamos a continuación:

Mundo real			**Mundo bootstrap**	
Distribución de probabilidad desconocida	Muestra aleatoria observada		Distribución empírica	Muestra bootstrap
$F \longrightarrow$	$\mathbf{x} = (x_1,...,x_n)$	$\Longrightarrow$	$\widehat{F} \longrightarrow$	$\mathbf{x}^* = (x_1^*,...,x_n^*)$
	$\downarrow$			$\downarrow$
	$\hat{\theta} = S(\mathbf{x})$			$\hat{\theta}^* = S(\mathbf{x}^*)$
	Estadístico de interés			Réplica bootstrap

En el mundo real partimos de una muestra observada $\mathbf{x} = (x_1, ..., x_n)$ obtenida por muestreo aleatorio de una distribución F. Deseamos estimar un parámetro de interés $\theta = T(F)$ en base a la muestra. Para este propósito, calculamos el estimador $\hat{\theta} = S(\mathbf{x})$ de dicho parámetro, y queremos conocer la distribución de probabilidad de ese estimador.

En el mundo bootstrap, la *muestra bootstrap* $\mathbf{x}^* = (x_1^*, ..., x_n^*)$ es obtenida por muestreo aleatorio de la distribución empírica $\widehat{F}$. A partir de esta muestra calcularemos $\hat{\theta}^* = S(\mathbf{x}^*)$, la *réplica bootstrap* del estadístico de interés. A partir de una muestra de valores de $\hat{\theta}^*$ se puede caracterizar la distribución de ese estimador.

La clave del bootstrap radica en el puente que une los dos mundos, en el proceso por el cuál construimos a partir de $\mathbf{x}$ un estimador $\widehat{F}$ para la distribución F.

Aunque la distribución F de X sea conocida, en general determinar la distribución exacta del estimador $\hat{\theta}$ es complicado. El método bootstrap permite estimar esta distribución mediante el siguiente algoritmo:

- Tomamos $B \in \mathbb{N}$ muestras bootstrap independientes $\mathbf{x}_1^*, ..., \mathbf{x}_B^*$ (en la siguiente sección veremos cómo obtenerlas).
- Para $b = 1, ..., B$ obtenemos la correspondiente réplica bootstrap del estimador $\hat{\theta}_b^* = S(\mathbf{x}_b^*)$.
- Estimamos la distribución muestral de $\hat{\theta}$ a través de su distribución empírica obtenida a partir de la muestra de las B réplicas bootstrap:

$$\widehat{\mathbb{P}}(\hat{\theta} \in A) = \mathbb{P}^*(\hat{\theta}^* \in A) = \frac{1}{B} \sum_{b=1}^{B} \mathbb{1}_A \left\{ \hat{\theta}_b^* \right\}.$$

Notar que calculamos un valor de $\hat{\theta} = S(\mathbf{x})$ pero que podemos calcular tantas réplicas bootstrap $\hat{\theta}^*$ como queramos, en este aspecto es donde radica la ventaja del bootstrap.

1.4. Obtención de muestras para aplicar el principio básico del bootstrap

Para la obtención de las muestras necesarias para aplicar el principio básico del bootstrap tendremos que atender a si conocemos o no la distribución de probabilidad F de la que proviene la muestra.

En caso afirmativo usaremos **métodos de Monte Carlo** y en caso contrario, hablaremos de **bootstrap paramétrico** y **bootstrap no paramétrico**. Cuando tengamos un modelo del que se conoce la familia de la distribución F pero no todos sus parámetros nos encontraremos en un modelo paramétrico. En caso contrario, si no hay un modelo ni hipótesis sobre la familia de F, el análisis estadístico será no paramétrico y se usará, únicamente, el hecho de que las variables aleatorias X_i son independientes e idénticamente distribuidas según una distribución F desconocida, será aquí donde la función empírica jugará un papel importante. El análisis no paramétrico puede completar al modelo paramétrico para la evaluación de la solidez de las conclusiones de un análisis paramétrico.

1.4.1. Métodos de Monte Carlo

Los métodos de Monte Carlo no son estrictamente métodos bootstrap al requerir que la distribución de probabilidad F de la que proviene la muestra sea conocida. Utilizando la distribución F se generan por simulación B muestras $\mathbf{x}_b$ independientes con $b = 1, ..., B$.

1.4.2. Bootstrap no paramétrico

El bootstrap no paramétrico también es denominado simplemente como bootstrap. Se basa en el remuestreo con reemplazamiento de la muestra original utilizando la distribución empírica.

Para obtener una muestra bootstrap, a partir del remuestreo de la muestra aleatoria $x_1,...,x_n$, elegiremos enteros $i_1,...,i_n$ que tomen valores entre 1 y n con la misma probabilidad $1/n$. De forma que cada muestra bootstrap $\mathbf{x}_b^*$ está formada por $x_{b,1}^* = x_{i_1},...,x_{b,n}^* = x_{i_n}$. Al realizar el muestreo con reemplazamiento garantizamos la independencia ya que cada observación está idénticamente distribuida según una distribución de probabilidad F y es independiente a las otras.

De esta forma se obtienen B muestras bootstrap independientes $x_{b,1}^*,...,x_{b,n}^* \overset{i.i.d}{\backsim} \widehat{F}$ para $b = 1,...,B$.

1.4.3. Bootstrap paramétrico

Cuando conozcamos la familia de distribución F pero no todos sus parámetros estimaremos estos últimos a partir de la muestra. La diferencia con el bootstrap no paramétrico es que no se realiza un muestreo con reemplazamiento sino que en este caso las muestras son obtenidas a partir de un estimador paramétrico de la función de distribución $\widehat{F}(\xi) = F(\hat{\xi})$, en lugar del estimador no paramétrico $\widehat{F}$. De esta forma, tenemos que $x_{b,1}^*,...,x_{b,n}^* \overset{i.i.d}{\backsim} F(\hat{\xi})$ para $b = 1,...,B$.

1.5. Estimaciones bootstrap del error estándar y del sesgo de $\hat{\theta}$

Una vez que hemos calculado un estimador $\hat{\theta}$, lo siguiente que queremos saber es la precisión y exactitud de ese estimador. El bootstrap supuso, acompañado de la evolución del cálculo computacional de los ordenadores, un notable avance en la obtención de estimaciones fiables del **error estándar** y del **sesgo**.

1.5.1. Estimación bootstrap del error estándar de un estimador

Siguiendo la notación de Efron y Tibshirani en [12] llamaremos *error estándar del estimador* $\hat{\theta}$ a la desviación típica de su distribución en el muestreo, es decir, $se_F(\hat{\theta}) = \sqrt{\mathbb{V}ar(\hat{\theta})}$.

Estimador bootstrap ideal del error estándar de $\hat{\theta}$

Presentamos este estimador con un ejemplo sencillo, el de la estimación del error estándar de la media muestral $\hat{\theta} = \overline{X}_n = \sum_{i=1}^{n} X_i/n$. En este caso $\mathbb{E}_F[\overline{X}_n] = \mu_F$ y $\mathbb{V}ar_F[\overline{X}_n] = \sigma_F^2/n$, por lo que, $se_F(\overline{X}_n) = \sigma_F/\sqrt{n}$. En esta situación, al tener una expresión explícita de $se_F(\overline{X}_n)$ en función de σ_F, podemos aplicar el principio plug-in directamente y obtener $\widehat{se}_F(\overline{X}_n) = \sigma_{\widehat{F}}/\sqrt{n}$.

Dado que el estimador plug-in de $\sigma_F = \sqrt{\mathbb{E}_F[(X-\mu_F)^2]}$ es $\sigma_{\widehat{F}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^2}$, ya que $\mu_{\widehat{F}} = \bar{x}$ y $\mathbb{E}_{\widehat{F}}[g(\mathbf{x})] = \frac{1}{n}\sum_{i=1}^{n} g(x_i)$ para cualquier función g, tenemos que

$$\widehat{se}_F(\overline{X}_n) = se_{\widehat{F}}(\overline{X}_n) = \frac{1}{n}\left[\sum_{i=1}^{n}(x_i-\bar{x})^2\right]^{1/2}.$$

Notar que hemos aplicado el principio plug-in en tres ocasiones, para estimar μ_F por $\mu_{\widehat{F}}$, σ_F por $\sigma_{\widehat{F}}$ y $se_F(\overline{X}_n)$ por $se_{\widehat{F}}(\overline{X}_n)$.

El estimador bootstrap obtenido con este procedimiento, es decir, el estimador plug-in que utiliza $\widehat{F}$ en lugar de F, $se_{\widehat{F}}(\hat{\theta})$, se denomina *estimador bootstrap ideal del error estándar de* $\hat{\theta}$. La limitación de

este procedimiento es que requiere una expresión explícita de $se(\hat{\theta})$ en función de σ_F, la varianza de X. Si no se dispone de una expresión de este tipo, es necesario recurrir a otros procedimientos bootstrap.

Estimador bootstrap del error estándar de $\hat{\theta}$

Este método se basa en obtener muestras bootstrap independientes de $\mathbf{x}$, con las que calcular réplicas $\hat{\theta}_b^*$ del estimador con $b = 1,...,B$. A partir de esta muestra de valores de $\hat{\theta}$, se puede estimar su error estándar $se_F(\hat{\theta})$ obteniendo el *estimador bootstrap del error estándar de* $\hat{\theta}$, $\widehat{se}_B(\hat{\theta})$. De forma más precisa, se utiliza el siguiente algoritmo:

- Seleccionamos B muestras bootstrap independientes $\mathbf{x}_1^*, \mathbf{x}_2^*, ..., \mathbf{x}_B^*$ cada una de n valores generadas por el remuestreo con reemplazamiento de $\mathbf{x}$.
- Evaluamos la réplica bootstrap del estimador en cada muestra bootstrap. Así, para $b = 1,...,B$ tenemos $\hat{\theta}_b^* = S(\mathbf{x}_b^*)$.
- Tenemos una muestra de B réplicas bootstrap $\hat{\theta}_1^*, ..., \hat{\theta}_B^*$ y estimamos el error estándar $se_F(\hat{\theta})$ con la desviación estándar muestral de la muestra de las B réplicas:

$$\widehat{se}_B(\hat{\theta}) = \left[\frac{1}{B-1}\sum_{b=1}^{B}(\hat{\theta}_b^* - \overline{\hat{\theta}^*})^2\right]^{1/2} \text{ donde } \overline{\hat{\theta}^*} = \frac{1}{B}\sum_{b=1}^{B}\hat{\theta}_b^*.$$

El estimador bootstrap del error estándar de $\hat{\theta}$ se aproxima al estimador bootstrap ideal del error estándar de $\hat{\theta}$ cuando el número de réplicas es suficientemente grande como vemos en [12, sección 6.2], es decir,

$$\lim_{B\to\infty} \widehat{se}_B(\hat{\theta}) = se_{\widehat{F}}(\hat{\theta}).$$

Elección del número de réplicas bootstrap

El estimador bootstrap ideal se obtiene cuando $B = \infty$. Sin embargo, debemos tener en cuenta que el coste computacional, el tiempo que se tarda en evaluar las B réplicas bootstrap, aumentará linealmente con B.

La estimación de la desviación típica de un estimador debe tener las mismas propiedades que cualquier otro estimador: poco sesgo y poca varianza. En un sentido asintótico, el estimador ideal $\widehat{se}_\infty(\hat{\theta})$ tiene la desviación estándar más pequeña posible entre los estimadores cuasi-insesgados de $se_F(\hat{\theta})$. Estas buenas propiedades se deben a que $\widehat{se}_\infty(\hat{\theta})$ es el estimador plug-in $se_{\widehat{F}}(\hat{\theta})$ como se afirma en [12, sección 6.4].

En esta misma sección, Efron y Tibshirani sugieren como reglas prácticas que incluso un número pequeño de réplicas como $B = 25$ resulta generalmente informativo. Así mismo, afirman que $B = 50$ es suficiente para obtener un buen estimador de $se_F(\hat{\theta})$ y que rara vez son necesarias más de 200 réplicas para estimar el error estándar.

1.5.2. Estimación bootstrap del sesgo de un estimador

El *sesgo de un estimador* $\hat{\theta} = S(\boldsymbol{x})$ es la diferencia entre la esperanza del estimador $\hat{\theta}$ y el valor real del parámetro θ, es decir,

$$\mathfrak{b}_F(\hat{\theta}) = \mathbb{E}_F[\hat{\theta}] - \theta.$$

Un sesgo alto no es una propiedad deseada en los estimadores. Cuando se cumpla $\mathbb{E}_F[\hat{\theta}] = \theta$, hablaremos de *estimador insesgado* y jugarán un papel importante en la estadística. Por lo general, los estimadores plug-in $T(\widehat{F})$ no son necesariamente insesgados, pero tienden a tener pequeños sesgos en comparación con sus errores estándar.

Estimador bootstrap ideal del sesgo de $\hat{\theta}$

El *estimador bootstrap ideal del sesgo de* $\hat{\theta}$ será la estimación $\mathfrak{b}_{\widehat{F}}(\hat{\theta})$ que obtenemos sustituyendo F por $\widehat{F}$ en la definición de sesgo, es decir, si $\theta = T(F)$, $\mathfrak{b}_{\widehat{F}}(\hat{\theta}) = \mathbb{E}_{\widehat{F}}[S(\mathbf{x}^*)] - T(\widehat{F})$. Aquí $T(\widehat{F})$, el estimador plug-in de θ, puede diferir de $\hat{\theta} = S(\mathbf{x})$. En otras palabras, $\mathfrak{b}_{\widehat{F}}(\hat{\theta})$ es el estimador plug-in de $\mathfrak{b}_F(\hat{\theta})$, independientemente de que $\hat{\theta}$ sea o no el estimador plug-in de θ.

Estimador bootstrap del sesgo de $\hat{\theta}$

Al igual que en el error estándar, aplicaremos el bootstrap para obtener las B réplicas que nos permitan obtener $\hat{\mathfrak{b}}_B(\hat{\theta})$, el *estimador bootstrap del sesgo de* $\hat{\theta}$, mediante el siguiente algoritmo:

Algoritmo bootstrap para el cálculo del estimador bootstrap del sesgo

- Seleccionamos B muestras bootstrap independientes $\mathbf{x}_1^*, \mathbf{x}_2^*, ..., \mathbf{x}_B^*$ cada una de n valores generadas por el remuestreo con reemplazamiento de $\mathbf{x}$.
- Evaluamos la réplica bootstrap de $\hat{\theta}$ correspondiente a cada muestra bootstrap, es decir, para $b = 1, ..., B$ tenemos $\hat{\theta}_b^* = S(\mathbf{x}_b^*)$.
- Estimamos $\mathbb{E}_F[S(\mathbf{x})]$ por el promedio $\overline{\hat{\theta}^*}$ y el estimador bootstrap del sesgo de $\hat{\theta}$ basado en las B réplicas es

$$\hat{\mathfrak{b}}_B(\hat{\theta}) = \overline{\hat{\theta}^*} - T(\widehat{F}). \tag{1.1}$$

Notar como el algoritmo bootstrap para la estimación del error estándar y para la estimación del sesgo difieren únicamente en el último paso, por tanto, podemos calcular tanto $\widehat{se}_B(\hat{\theta})$ como $\hat{\mathfrak{b}}_B(\hat{\theta})$ con el mismo conjunto de réplicas bootstrap.

Estimador mejorado del sesgo

Podemos ver un método mejor, desarrollado en [12, sección 10.4], para aproximar el sesgo cuando $\hat{\theta}$ es el estimador plug-in $T(\widehat{F})$ de $\theta = T(F)$. Dada una muestra $\mathbf{x} = (x_1, ..., x_n)$ de n valores distintos, obtenemos por remuestreo una muestra bootstrap $\mathbf{x}^* = (x_1^*, ..., x_n^*)$ como hemos visto anteriormente. Comenzamos denotando por $P_j^* = \#\{x_i^* = x_j\}/n$ la frecuencia en la que aparece la componente x_j en la muestra bootstrap para $j = 1, ..., n$. Y llamaremos *vector de remuestreo* a $\mathbf{P}^* = (P_1^*, ..., P_n^*)$, notar que todas las componentes son no negativas y que suman 1.

Podemos escribir la réplica bootstrap $\hat{\theta}^* = S(\mathbf{x}^*)$ en función del vector de remuestreo $\mathbf{P}^*$. Por ejemplo, para el caso de la media muestral cuando $\hat{\theta} = \overline{X}_n$ podemos escribir $\hat{\theta}^* = \sum_{j=1}^{n} P_j^* x_j$. De esta forma, fijamos los valores de $\mathbf{x}$ y serán las componentes P_j^* las que variarán aleatoriamente. Escribiremos $\hat{\theta}^* = \Gamma(\mathbf{P}^*)$ denotando que $\hat{\theta}^*$ es función del vector de remuestreo $\mathbf{P}^*$.

Denotaremos por $\mathbf{P}^0$ al vector de tamaño n cuyas entradas son todas $1/n$, es decir, $\mathbf{P}^0 = (1/n, ..., 1/n)$. Notar que $\Gamma(\mathbf{P}^0)$ es el estimador $\hat{\theta}^*$ donde $P_j^* = 1/n$ para todo $j = 1, ..., n$, es decir, cada dato x_j de la muestra original aparece exactamente una única vez en la muestra bootstrap $\mathbf{x}^*$. Notar que salvo permutaciones en el orden, tenemos que $\mathbf{x}^* = \mathbf{x}$ y el estadístico $\hat{\theta}$ no cambia en este caso pues $\widehat{F}$ tampoco cambia, es decir, $\Gamma(\mathbf{P}^0) = \hat{\theta} = T(\widehat{F})$, el valor del estimador observado en la muestra original.

Dadas las B muestras bootstrap $\mathbf{x}_1^*, ..., \mathbf{x}_B^*$ y sus correspondientes vectores de remuestreo $\mathbf{P}_1^*, ..., \mathbf{P}_B^*$ denotamos por $\overline{\mathbf{P}}^* = \frac{1}{B} \sum_{b=1}^{B} \mathbf{P}_b^*$ la media de dichos vectores. Entonces la estimación bootstrap del sesgo en (1.1) es

$$\hat{\mathfrak{b}}_B(\hat{\theta}) = \overline{\hat{\theta}^*} - \Gamma(\mathbf{P}^0),$$

y el *mejor estimador bootstrap del sesgo de* $\hat{\theta}$ será

$$\overline{\overline{\mathfrak{b}}}_B(\hat{\theta}) = \overline{\hat{\theta}^*} - \Gamma(\overline{\mathbf{P}}^*).$$

Ambas estimaciones $\hat{\mathfrak{b}}_B(\hat{\theta})$ y $\overline{\overline{\mathfrak{b}}}_B(\hat{\theta})$ convergerán asintóticamente con $B \to \infty$ al estimador bootstrap ideal del sesgo, es decir, $\hat{\mathfrak{b}}_\infty(\hat{\theta}) = \overline{\overline{\mathfrak{b}}}_\infty(\hat{\theta}) = \mathfrak{b}_{\widehat{F}}(\hat{\theta})$, siendo la convergencia más rápida para $\overline{\overline{\mathfrak{b}}}_B(\hat{\theta})$. Efron y Tibshirani muestran en [12, sección 23.4] que $\overline{\overline{\mathfrak{b}}}_B(\hat{\theta})$ equivale a usar $\hat{\mathfrak{b}}_{CB}(\hat{\theta})$ donde C es una constante generalmente mayor que 50.

Corrección del sesgo

Generalmente la finalidad de estimar el sesgo de un estimador $\hat{\theta}$ es corregir dicho estimador de modo que sea menos sesgado. Si el estimador bootstrap del sesgo es $\hat{\mathfrak{b}}_B(\hat{\theta}) = \overline{\hat{\theta}^*} - \hat{\theta}$ entonces el *estimador de* θ *con sesgo corregido* es definido por

$$\hat{\theta}_c = \hat{\theta} - \hat{\mathfrak{b}}_B(\hat{\theta}) = 2\hat{\theta} - \overline{\hat{\theta}^*}.$$

Notar que si $\overline{\hat{\theta}^*}$ es mayor que $\hat{\theta}$, entonces $\hat{\theta}_c$ será menor que $\hat{\theta}$.

La corrección del sesgo no siempre es conveniente debido a la alta variabilidad de la estimación del sesgo que puede conllevar un error estándar alto en el estimador corregido como puede verse en [12, sección 10.6]. Así, si $\hat{\mathfrak{b}}_B(\hat{\theta})$ es pequeño comparado con $\widehat{se}_B(\hat{\theta})$, entonces es seguro usar $\hat{\theta}$ en lugar de $\hat{\theta}_c$. El caso contrario puede ser un indicador de que el estadístico $\hat{\theta} = S(\mathbf{x})$ no es una estimación apropiada del parámetro θ.

1.6. Método Jackknife

El *jackknife*, considerado predecesor del bootstrap, es el método computacional desarrollado por Maurice Quenouille en 1949 inicialmente con el objetivo de mejorar una estimación al corregir el sesgo. Más tarde, se descubrió que aún era más útil como forma de estimar errores estándar de los estimadores.

Dada la muestra $\mathbf{x} = (x_1, ..., x_n)$, llamaremos *muestra i-ésima jackknife* a $\mathbf{x}_{(i)} = (x_1, ..., x_{i-1}, x_{i+1}, ..., x_n)$ para $i = 1, ..., n$ y *réplica i-ésima jackknife* del estadístico $\hat{\theta} = S(\mathbf{x})$ a $\hat{\theta}_{(i)} = S(\mathbf{x}_{(i)})$ para $i = 1, ..., n$. Así, en lugar de obtener las muestras bootstrap por remuestreo con reemplazamiento, el jackknife se centra en las n muestras fijas $\mathbf{x}_{(1)}, ..., \mathbf{x}_{(n)}$ obtenidas eliminando la i-ésima observación.

1.6.1. Estimación jackknife del error estándar de un estimador

Al igual que en el bootstrap, la estimación del error estándar de un estimador se basará en la desviación estándar, en este caso, de las n muestras obtenidas tras eliminar la i-ésima componente. Por tanto el *estimador jackknife del error estándar de* $\hat{\theta}$, donde $\hat{\theta}_{(\cdot)} = \dfrac{1}{n}\sum_{i=1}^{n} \hat{\theta}_{(i)}$ es el promedio de las réplicas, será

$$\widehat{se}_J(\hat{\theta}) = \left[\frac{n-1}{n}\sum_{i=1}^{n}(\hat{\theta}_{(i)} - \hat{\theta}_{(\cdot)})^2\right]^{1/2}.$$

Notar que el factor $\dfrac{n-1}{n}$ es mucho mayor que el correspondiente al bootstrap $\dfrac{1}{B-1}$. Este factor se debe intuitivamente a que las desviaciones en el caso del jackknife $(\hat{\theta}_{(i)} - \hat{\theta}_{(\cdot)})^2$ tienden a ser más pequeñas que las del bootstrap $(\hat{\theta}^*_b - \overline{\hat{\theta}^*})^2$.

La elección de $\frac{n-1}{n}$ es convencionalmente arbitraria. Como puede verse en [12, sección 11.2], si consideramos el caso de la media muestral, $\hat{\theta} = \overline{X}_n$, tenemos que

$$\hat{\theta} = \overline{X}_n \ , \ \hat{\theta}_{(i)} = \frac{n\hat{\theta} - x_i}{n-1} \ , \ \hat{\theta}_{(\cdot)} = \hat{\theta} \ , \ \hat{\theta}_{(i)} - \hat{\theta}_{(\cdot)} = \frac{\overline{x} - x_i}{n-1}.$$

Y, sustituyendo, la expresión del estimador jackknife del error estándar de la media muestral es

$$\widehat{se}_J(\hat{\theta}) = \left[\frac{1}{n(n-1)} \sum_{i=1}^{n} (x_i - \overline{x})^2 \right]^{1/2},$$

es decir, es igual al estimador insesgado del error estándar de la media muestral.

Si consideráramos un factor $\left[\frac{n-1}{n}\right]^2$, entonces obtendríamos la estimación plug-in del error estándar de la media muestral,

$$\widehat{se}_J(\overline{X}_n) = \frac{1}{n} \left[\sum_{i=1}^{n} (x_i - \overline{x})^2 \right]^{1/2} = \widehat{se}_F(\overline{X}_n),$$

pero esta estimación no diferirá sustancialmente de la insesgada a menos que n sea pequeño.

1.6.2. Estimación jackknife del sesgo de un estimador

El *estimador jackknife del sesgo de* $\hat{\theta}$ es un múltiplo del promedio de las desviaciones jackknife o *valores jackknife de influencia*, $\hat{\theta}_{(i)} - \hat{\theta}$ para $i = 1, ..., n$, siendo dicho estimador

$$\hat{\mathfrak{b}}_J(\hat{\theta}) = (n-1)(\hat{\theta}_{(\cdot)} - \hat{\theta}).$$

El factor $n-1$ es convencionalmente arbitrario y se debe al considerar en este caso, como puede verse también en [12, sección 11.2], la varianza muestral

$$\hat{\theta} = \sum_{i=1}^{n} (x_i - \overline{x})^2 / n.$$

El sesgo de la varianza muestral es $-\sigma^2/n$, es decir, $-1/n$ veces la varianza poblacional. Y, si calculamos la expresión del estimador jackknife del sesgo de la varianza muestral, obtenemos que

$$\hat{\mathfrak{b}}_J(\hat{\theta}) = -\frac{1}{n(n-1)} \sum_{i=1}^{n} (x_i - \overline{x})^2,$$

es decir, es $-1/n$ veces el estimador insesgado de la varianza poblacional, $\sum (x_i - \overline{x})^2/(n-1)$.

Notar que si, en lugar de la varianza muestral, hubiésemos elegido la media muestral como hemos hecho para el error estándar, los valores jackknife de influencia serían cero al ser insesgada la media muestral.

1.6.3. Relación entre el jackknife y el bootstrap

Hemos visto dos métodos para la estimación del error estándar y del sesgo de un estimador. El jackknife es una aproximación del bootstrap salvo en algunos tipos de estimadores.

En efecto, consideremos un estadístico lineal, es decir, que puede expresarse de la forma

$$\hat{\theta} = S(\mathbf{x}) = \mu + \frac{1}{n} \sum_{i=1}^{n} \alpha(x_i),$$

donde μ es una constante y $\alpha(\cdot)$ una función.

En [12, sección 11.9] se sigue que el estimador bootstrap del error estándar de $\hat{\theta}$ es

$$\left[\frac{1}{n^2}\sum_{i=1}^{n}(\alpha_i-\overline{\alpha})^2\right]^{1/2},$$

mientras que el estimador jackknife del error estándar de $\hat{\theta}$ es

$$\left[\frac{1}{n(n-1)}\sum_{i=1}^{n}(\alpha_i-\overline{\alpha})^2\right]^{1/2},$$

donde $\alpha_i = \alpha(x_i)$.

Vemos que los dos estimadores del error estándar coinciden salvo por un pequeño factor $\sqrt{\frac{n-1}{n}}$. Por tanto, como se afirma en [12, sección 11.6], para un estadístico lineal no hay pérdida de información al usar jackknife ya que el conocimiento de un estadístico lineal para las n muestras jackknife $\mathbf{x}_{(i)}$ determina el valor de $\hat{\theta}$ para cualquier conjunto de muestras bootstrap $\mathbf{x}^*$.

Cuando el estadístico $\hat{\theta}$ es no lineal, hay pérdida de información. Siguiendo los pasos recogidos en [12, sección 11.9], podemos aproximar el estadístico no lineal $\hat{\theta}$ por uno lineal $\hat{\theta}_{lin}$ tal que $\hat{\theta}_{lin}$ tiene el mismo valor que $\hat{\theta}$ para las muestras jackknife $\mathbf{x}_{(i)}$. De esta forma, la estimación jackknife del error estándar de $\hat{\theta}$ coincide con la estimación bootstrap del error estándar de $\hat{\theta}_{lin}$, excepto por el pequeño factor anterior $\sqrt{\frac{n-1}{n}}$. Por tanto, la precisión del estimador jackknife del error estándar de $\hat{\theta}$ dependerá de cómo de lineal sea $\hat{\theta}$. Para aquellos que sean poco lineales, el jackknife puede ser muy ineficiente.

Capítulo 2

Intervalos de confianza

Parte del capítulo anterior se ha centrado en el cálculo de errores estándar mediante procedimientos bootstrap. En este capítulo veremos como el error estándar es usado frecuentemente para construir intervalos de confianza de un parámetro θ de interés. La evaluación de la incertidumbre de los valores de los parámetros se realizará utilizando estos intervalos permitiendo determinar el grado de error asociado a la estimación del parámetro.

En este capítulo veremos diferentes técnicas para la construcción de intervalos de confianza. En primer lugar, analizaremos como la teoría clásica de construcción de intervalos a través de la tipificación a una distribución Normal estándar puede utilizarse para obtener resultados mediante procedimientos bootstrap. A continuación, veremos como el bootstrap mejora los intervalos de confianza sin necesidad de asumir una distribución normal.

2.1. Intervalos de confianza bootstrap estándar

2.1.1. Intervalos de confianza basados en teoría clásica

Supongamos que se tiene una muestra $\mathbf{x}$ de tamaño n obtenida por muestreo aleatorio con una distribución F desconocida como en el capítulo anterior. Sea $\hat{\theta}$ el estimador del parámetro de interés $\theta = T(F)$.

Bajo numerosas circunstancias, como en el caso de estimadores máximo verosímiles, a medida que el tamaño muestral n crece, la distribución muestral de $\hat{\theta}$ se vuelve asintóticamente normal, con media θ y varianza $se^2(\theta)$, generalmente desconocida y que debe estimarse por $\widehat{se}(\hat{\theta})$. Es decir,

$$\hat{\theta} \dot{\sim} \mathcal{N}(\theta, \widehat{se}(\hat{\theta})) \text{ o equivalentemente } Z = \frac{\hat{\theta} - \theta}{\widehat{se}(\hat{\theta})} \dot{\sim} \mathcal{N}(0,1).$$

Entonces $\mathbb{P}\left(z_\alpha \leq (\hat{\theta} - \theta)/\widehat{se}(\hat{\theta}) \leq z_{1-\alpha}\right) = 1 - 2\alpha$, donde $\alpha \in [0,1]$ y z_α es el $100 \cdot \alpha$-ésimo percentil de la distribución $\mathcal{N}(0,1)$.

Y obtenemos el siguiente intervalo para θ:

$$\left[\hat{\theta} - z_{1-\alpha} \cdot \widehat{se}(\hat{\theta}); \hat{\theta} - z_\alpha \cdot \widehat{se}(\hat{\theta})\right],$$

que llamaremos *intervalo de confianza estándar con nivel de confianza del* $100 \cdot (1 - 2\alpha)\,\%$. El nivel de confianza indica que con probabilidad $100 \cdot (1 - 2\alpha)\,\%$ el intervalo contendrá el valor verdadero de θ. [1]

[1] En lugar del tratamiento usual de intervalos de confianza con un nivel de confianza $1 - \alpha$, seguimos la notación de Efron y Tibshirani en [12]. Consideraremos intervalos de nivel de confianza $1 - 2\alpha$. Notar que simplemente se aplica un cambio de notación sin que los resultados se vean alterados.

Además, teniendo en cuenta que $z_\alpha = -z_{1-\alpha}$, podemos expresar el intervalo de la siguiente forma:

$$\hat{\theta} \mp z_{1-\alpha} \cdot \widehat{se}(\hat{\theta}).$$

La aproximación anterior es válida para muestras finitas cuando n sea suficientemente grande. Cuando las muestras sean pequeñas podemos mejorar los intervalos asintóticos usando intervalos t de Student. Para el caso $\hat{\theta} = \overline{X}_n$ (Gosset, 1908):

$$\frac{\hat{\theta} - \theta}{\widehat{se}(\hat{\theta})} \overset{\cdot}{\backsim} t_{n-1},$$

donde t_{n-1} es la distribución t de Student con $n-1$ grados de libertad.

A partir de esta aproximación obtenemos el intervalo $\left[\hat{\theta} - t_{n-1;1-\alpha} \cdot \widehat{se}(\hat{\theta}), \hat{\theta} - t_{n-1;\alpha} \cdot \widehat{se}(\hat{\theta})\right]$, donde $t_{n-1;\alpha}$ es el α-ésimo percentil de la distribución t de $n-1$ grados de libertad.

Efron y Tibshirani observan en [12, sección 12.4] que, para este caso, la distribución es exacta si las observaciones se distribuyen normalmente y tiene el efecto de ampliar el intervalo para ajustar el hecho de que el error estándar sea desconocido. No obstante, cuando $n \geq 20$, los percentiles de la distribución t no difieren mucho de los de la distribución $\mathcal{N}(0,1)$.

2.1.2. Intervalos de confianza clásicos bootstrap

La metodología clásica del cálculo de intervalos de confianza explicada en la sección previa se puede combinar con la metodología bootstrap del capítulo anterior para obtener estimadores del error estándar $se(\hat{\theta})$, dando lugar a los **intervalos de confianza bootstrap estándar**. Es importante señalar que este procedimiento requiere la normalidad o normalidad asintótica de los estimadores.

2.2. Intervalos bootstrap-t

Los **intervalos bootstrap-t** mejoran la aproximación anterior ya que a través del bootstrap podemos prescindir de las suposiciones de normalidad de los estimadores. La idea es calcular una tabla de percentiles a partir de los datos que tenemos y sobre la que construir el intervalo de confianza de la misma manera que en el caso de los intervalos tradicionales. De esta forma, generaremos B muestras bootstrap y calcularemos la versión bootstrap de Z para cada una de ellas como se ve en el siguiente algoritmo:

- Generamos B muestras bootstrap $\mathbf{x}_1^*, ..., \mathbf{x}_B^*$.
- Por cada muestra calculamos

  $$Z_b^* = \frac{\hat{\theta}_b^* - \hat{\theta}}{\widehat{se}_b^*} \text{ para } b = 1, ..., B,$$

  donde $\hat{\theta}_b^* = S(\mathbf{x}_b^*)$ y $\widehat{se}_b^*$ son respectivamente el valor de $\hat{\theta}$ y la estimación del error estándar de $\hat{\theta}^*$ para la muestra bootstrap $\mathbf{x}_b^*$.
- El α-ésimo percentil de Z_b^* es estimado por el valor $\hat{t}_\alpha$ tal que $\#\left\{Z_b^* \leq \hat{t}_\alpha\right\}/B = \alpha$. Por ejemplo, si $B = 1000$, el estimador del punto 5 % es el 50-ésimo valor mayor de los Z_b^* ordenados y el estimador del punto 95 % es el 950-ésimo valor mayor de los Z_b^* ordenados.
- El *intervalo bootstrap-t* es $\left[\hat{\theta} - \hat{t}_{1-\alpha} \cdot \widehat{se}(\hat{\theta}), \hat{\theta} - \hat{t}_\alpha \cdot \widehat{se}(\hat{\theta})\right]$.

Si $B \cdot \alpha$ no es un entero, entonces seguiremos el procedimiento propuesto en [12, sección 12.5]:

- Asumamos que $\alpha \leq 0{,}5$ y sea $k = \lceil (B+1)\alpha \rceil$, el mayor entero menor o igual que $(B+1)\alpha$.

- Definimos los α y $1-\alpha$-ésimos percentiles como los k y $(B+1-k)$-ésimo valores mayores de Z_b^* respectivamente.

Es importante notar que para la construcción de intervalos de confianza no es suficiente con tomar $B = 100$ o 200, es necesario un mayor número de réplicas bootstrap. Efron y Tibshirani en [12, sección 19.3] afirman que B debería ser mayor de 500. Se necesita un mayor número de muestras bootstrap para estimar el percentil 95, frente a las necesarias para estimar el error estándar, porque el percentil depende de la cola de la distribución en la que se producen menos muestras. En otras palabras, los estadísticos bootstrap que dependen de las colas extremas de la distribución de $\hat{\theta}^*$ requerirán un mayor número de muestras bootstrap para alcanzar una precisión aceptable.

Ventajas y desventajas

Los percentiles de la distribución normal o la t de Student son simétricos alrededor del cero, lo que permite obtener intervalos simétricos alrededor del estimador del parámetro $\hat{\theta}$. Por el contrario, los percentiles bootstrap-t pueden ser asimétricos alrededor del cero provocando intervalos más largos o cortos a derecha o izquierda. Esta asimetría supone una mejora en el cubrimiento que tienen estos intervalos. El procedimiento bootstrap-t solo es una buena opción para estadísticos de localización como la media muestral o la mediana.

Existen algunos problemas de cálculo y de interpretación en los intervalos bootstrap-t. En el denominador del estadístico Z_b^* se necesita conocer $\widehat{se}_b^*$, es decir, la desviación estándar de $\hat{\theta}^*$ para cada muestra bootstrap $\mathbf{x}_b^*$ con $b = 1,...,B$. Sin embargo, para muchos estadísticos no dispondremos de una expresión explícita. Ello nos obligará a calcular un estimador bootstrap del error estándar para cada muestra bootstrap suponiendo un importante coste computacional.

Los intervalos bootstrap-t pueden conducir a intervalos que a menudo son demasiado amplios y quedan fuera del rango permitido para un parámetro si no se aplica alguna transformación a dicho parámetro. Por ejemplo, si consideramos θ como el coeficiente de correlación poblacional y construimos un intervalo de confianza para la transformación de Fisher $\phi = \frac{1}{2} log\left(\frac{1+\theta}{1-\theta}\right)$, que es apropiada para datos con una distribución normal bivariante, y aplicamos la transformación inversa $(e^{2\phi}-1)/(e^{2\phi}+1)$ a los extremos del intervalo, entonces obtenemos un intervalo mejor para θ. Si consideramos los datos que pueden verse en [12, sección 3.2] y calculamos el intervalo de confianza bootstrap-t con un nivel del 90 % para ϕ y aplicamos la transformación inversa, obtenemos el intervalo $[0{,}45; 0{,}93]$ para θ, que es más pequeño que el obtenido sin transformación $[-0{,}026; 0{,}90]$. Si calculamos el intervalo con un nivel del 98 %, obtenemos el intervalo $[0{,}17; 0{,}95]$ si usamos la transformación y $[-0{,}66; 1{,}03]$ si no la usamos. Notar que este último intervalo queda fuera de los valores posibles del coeficiente de correlación.

Por tanto, los intervalos bootstrap-t no son invariantes cuando se aplican transformaciones al parámetro que se estima y para la mayoría de los problemas, no sabremos qué transformación es mejor aplicar, y esto será un gran inconveniente para el uso general de bootstrap-t para la construcción del intervalo de confianza.

2.3. Intervalos bootstrap de tipo percentil

Otro enfoque para calcular intervalos de confianza bootstrap se basa en los **percentiles** de la distribución bootstrap del estadístico.

En el intervalo de confianza basado en la normalidad $\hat{\theta} \mp z_{1-\alpha} \cdot \widehat{se}(\hat{\theta})$, los extremos del intervalo son los percentiles $100 \cdot \alpha$ y $100 \cdot (1-\alpha)$ de la distribución de $\hat{\theta}^* = S(\mathbf{x}^*)$, es decir, $\mathcal{N}(\hat{\theta}, \widehat{se}(\hat{\theta}))$. La idea del intervalo de confianza bootstrap de percentiles es estimar esos percentiles de otra forma que veremos a continuación.

Sea $\widehat{G}$ la función de distribución empírica de la muestra. El *intervalo de confianza tipo percentil de nivel* $1-2\alpha$ está definido por los percentiles α y $1-\alpha$ de la distribución de $\hat{\theta}^*$, que se pueden expresar en términos de $\widehat{G}$ donde $\hat{\theta}^*_\alpha = \widehat{G}^{-1}(\alpha)$. Por lo que el intervalo de confianza es $\left[\widehat{G}^{-1}(\alpha), \widehat{G}^{-1}(1-\alpha)\right]$.

Estas expresiones corresponden a la situación ideal en la que disponemos de infinitas réplicas bootstrap. Pero en la práctica tomaremos un número finito $B \in \mathbb{N}$ siendo el procedimiento el siguiente:

- Generamos B muestras bootstrap $\mathbf{x}^*_1, ..., \mathbf{x}^*_B$.
- Por cada muestra calculamos los respectivos estimadores $\hat{\theta}^*_b = S(\mathbf{x}^*_b)$ con $b = 1, ..., B$.
- Denotamos por $\hat{\theta}^*_{B(\alpha)}$ el percentil $100 \cdot \alpha$-ésimo de los valores $\hat{\theta}^*_b$, es decir, el $B \cdot \alpha$-ésimo valor en la muestra ordenada de las B réplicas de $\hat{\theta}^*$. Por ejemplo, si $B = 1000$ y $\alpha = 0{,}05$, entonces $\hat{\theta}^*_{B(\alpha)}$ es el 50-ésimo valor de la muestra ordenada de las réplicas. Si $B \cdot \alpha$ no es un entero, seguiremos el convenio descrito en la sección anterior.
- Análogamente obtenemos que $\hat{\theta}^*_{B(1-\alpha)}$ es el percentil $100 \cdot (1-\alpha)$ de los valores ordenados.
- El *intervalo bootstrap tipo percentil* $1-2\alpha$ es $\left[\hat{\theta}^*_{B(\alpha)}, \hat{\theta}^*_{B(1-\alpha)}\right]$.

Si la distribución de $\hat{\theta}^*$ es aproximadamente normal, entonces los intervalos estándar normal y tipo percentil coincidirán prácticamente.

Transformaciones del intervalo percentil

Cuando la distribución del estimador no sea normal o asintóticamente normal, podemos realizar una transformación del estimador para normalizarlo. De esta forma, podremos aplicar la teoría estándar para la construcción de intervalos de confianza que hemos visto en la sección [2.1]. Sin embargo, encontrar la transformación que nos lleve a ese nuevo estimador no siempre será evidente.

El intervalo de percentiles, que no requiere normalidad, es una herramienta útil como podemos ver en el siguiente ejemplo recogido en [12, sección 13.3] y del que desarrollamos el código R en la sección [A.2] del Apéndice:

Supongamos que se genera una muestra $\mathbf{x}$ de tamaño 10 de una distribución normal estándar. Tomamos como parámetro de interés, que deseamos estimar, $\theta = e^\mu$ donde μ es la media poblacional. El verdadero valor de θ es $e^0 = 1$ mientras que el valor de $\hat{\theta} = e^{\bar{x}}$ es 1,25.

Si se generan 1000 réplicas bootstrap entonces se obtiene una distribución bastante asimétrica y el intervalo percentil con un nivel del 95 % es $[0{,}75; 2{,}07]$. Obtenemos que la estimación bootstrap del error estándar es $\widehat{se}_{1000}(\hat{\theta}) = 0{,}34$. Por lo que el intervalo bootstrap estándar es $1{,}25 \pm 1{,}96 \cdot 0{,}34 = [0{,}59; 1{,}92]$. Notar las discrepancias entre los dos intervalos debidas a que la distribución del estadístico no es normal.

Consideremos la transformación logaritmo sobre los datos, $\phi = log(\theta)$, que mejora la simetría de la distribución de $\hat{\theta}$ haciéndola normal. Debido a esa normalidad el intervalo bootstrap estándar y el de percentiles casi coinciden, siendo $[-0{,}28; 0{,}73]$ y $[-0{,}29; 0{,}73]$ respectivamente. Notar que esto es razonable ya que $\hat{\phi}^* = \bar{x}^*$.

Por tanto, parece lógico basar el intervalo bootstrap estándar en $\widehat{\phi}$, y luego transformarlo a un intervalo de confianza para θ, en lugar de basarlo directamente en $\hat{\theta}$. El inverso de la aplicación logaritmo es la exponencial. Por lo que tomando la exponencial del intervalo obtenido para ϕ llegamos a $[0{,}76; 2{,}08]$ que es más próximo al intervalo percentil calculado al principio $[0{,}75; 2{,}07]$ que al intervalo estándar $[0{,}59; 1{,}92]$ construido usando $\hat{\theta}$ directamente.

Capítulo 3

Contrastes de hipótesis

El objetivo del contraste de hipótesis es decidir, basándose en la información proporcionada por la muestra observada $\mathbf{x}$, entre dos afirmaciones mutuamente excluyentes relativas a la distribución de probabilidad de la población. Estas afirmaciones se denominan *hipótesis nula* e *hipótesis alternativa* y las denotaremos por H_0 y H_1 respectivamente.

Bajo la hipótesis nula, la distribución de probabilidad de la muestra $\mathbf{x}$ podrá quedar totalmente especificada o, por el contrario, algún parámetro o la familia de la distribución será desconocido. En este capítulo vamos a ver algunas de las diferentes alternativas que existen para poder afrontar estas situaciones.

3.1. Elementos básicos de un contraste

Un contraste de hipótesis comprende los elementos que esquematizamos a continuación:

- La hipótesis nula. La situación más sencilla implica una **hipótesis nula simple** H_0 que especifica completamente la distribución de probabilidad de los datos. Es decir, tenemos una única muestra $\mathbf{x} = (x_1, ..., x_n)$ de una población con función de distribución F. Entonces H_0 especifica $F = F_0$, donde F_0 no contiene parámetros desconocidos.

  Sin embargo, lo más usual será que tengamos una **hipótesis nula compuesta** en la que algunos aspectos de F no estén determinados y permanezcan desconocidos cuando H_0 sea cierta.

  Ejemplos respectivos de estas situaciones son una exponencial de media 1 y una normal de media 1 donde la varianza permanece inespecificada.

  Una hipótesis es una afirmación sobre la distribución de la población, habitualmente sobre uno de sus parámetros. Dada una partición del espacio paramétrico $\Theta = \Theta_0 \cup \Theta_1$, la hipótesis nula H_0 afirmará que $\theta \in \Theta_0$ y la hipótesis alternativa H_1 que $\theta \in \Theta_1$. De esta forma, el contraste de hipótesis se muestra como una regla de decisión que, a cada posible muestra de $\mathbf{x}$, le asigna el rechazo o no rechazo de H_0.

- Un estadístico $T \equiv T(X)$, llamado *estadístico test*, cuyo valor observado es $t \equiv T(\mathbf{x})$.

- Una *región crítica o de rechazo C*. Si el valor del estadístico test en la muestra pertenece a la región, entonces rechazamos la hipótesis nula H_0.

Debemos tener en cuenta que el contraste de hipótesis es un método estadístico de inferencia, que no demuestra la validez o certeza de la hipótesis que se acepte. Debe entenderse que la información disponible proporciona, o no, evidencia suficiente, según el criterio que fijemos, en contra de la hipótesis nula H_0; en el primer caso se rechazará H_0 y en el segundo no se rechazará.

Como la decisión entre H_0 y H_1 se toma a partir de la información de la muestra observada $\mathbf{x}$, el contraste tiene un error asociado. Se pueden cometer dos tipos de error: cuando siendo cierta H_0 se

rechace, cometeremos un *error de tipo I* y cuando siendo falsa no se rechace, cometeremos un *error de tipo II*.

Llamaremos *nivel de significación* α *de un test* a la probabilidad de cometer un error de tipo I. Vendrá dado por el supremo de estas probabilidades entre todas las distribuciones de probabilidad que satisfacen la hipótesis nula, es decir,

$$\alpha = \sup_{\theta \in \Theta_0} \mathbb{P}_\theta(X \in C) = \sup_{\theta \in \Theta_0} \int_C f_\theta(x)dx.$$

Usualmente α es un valor pequeño fijo, por ejemplo, $\alpha = 0{,}05$ o $\alpha = 0{,}01$.

En la mayoría de los contrastes, el test y la región crítica se definen a partir del estadístico test. La determinación de la región crítica dependerá de cómo se establezca la hipótesis alternativa:

- $C = \{\mathbf{x} : t > c_1 \text{ o } t < c_2\}$ si el contraste es bilateral ($H_1 : \theta \neq \theta_0$).
- $C = \{\mathbf{x} : t > c_3\}$ si el contraste es unilateral derecho ($H_1 : \theta > \theta_0$).
- $C = \{\mathbf{x} : t < c_4\}$ si el contraste es unilateral izquierdo ($H_1 : \theta < \theta_0$).

donde llamaremos a c_i para $i = 1,2,3,4$ el *valor crítico*. El problema será cómo elegir c_i con el fin de lograr el nivel de significación prefijado α. Para ello se elige como valor crítico al percentil correspondiente de la distribución del estadístico, es decir, $c_1 = z_{1-\alpha/2}$, $c_2 = z_{\alpha/2}$, $c_3 = z_{1-\alpha}$, $c_4 = z_\alpha$.

Una herramienta alternativa para establecer una regla de decisión es definir el *p-valor* como la probabilidad, bajo H_0, de obtener un valor tan extremo o más como el que hemos observado. Por ejemplo, para el contraste unilateral derecho será la probabilidad de observar un valor del estadístico mayor que el que hemos observado siendo cierta H_0, es decir,

$$p = \mathbb{P}(T \geq t | H_0). \tag{3.1}$$

El siguiente lema, enunciado en [1, sección 1.1], nos permitirá determinar la distribución del p-valor como variable aleatoria:

Lema 3.1. *Si X es una variable aleatoria con distribución de probabilidad continua F, entonces la variable aleatoria* $1 - F(X) \sim \mathcal{U}(0,1)$.

Por tanto, si F_{T_0} es la función de distribución de T bajo la hipótesis nula, entonces $P = 1 - F_{T_0}(T)$ tiene una distribución uniforme bajo H_0. Para realizar un contraste de hipótesis, simplemente calculamos el p-valor y rechazamos H_0 si el valor es menor que α.

3.2. Contrastes de hipótesis nula simple

Si H_0 es simple, la distribución de los datos bajo la hipótesis nula está completamente especificada y el nivel de significación es $\alpha = \mathbb{P}(X \in C | H_0)$. En estas situaciones donde la distribución de la muestra es completamente conocida bajo H_0, se pueden utilizar métodos de tipo Monte Carlo para calcular el p-valor e implementar un contraste.

3.2.1. Contrastes de Monte Carlo

Cuando conocemos la distribución de la muestra, podremos generar nuevas muestras mediante métodos de Monte Carlo como hemos visto en la sección [1.4.1] con la salvedad de que aquí el modelo de simulación deberá satisfacer la hipótesis nula. A partir de estas muestras, calcularemos el valor del estadístico para cada una. Y con la muestra formada por los valores del estadístico podremos estimar el p-valor como vemos en el siguiente algoritmo para los contrastes de Monte Carlo de hipótesis simple:

- Utilizando la distribución especificada por H_0, generamos B muestras $\mathbf{x}_1, ..., \mathbf{x}_B$.
- Evaluamos el estadístico correspondiente a cada muestra, es decir, $t_b^* = T(\mathbf{x}_b)$ para $b = 1, ..., B$.
- Calculamos $\hat{p} = \frac{1}{B} \sum_{b=1}^{B} \mathbb{1}\{t_b^* \geq t\}$ y rechazamos H_0 si $\hat{p} \leq \alpha$.

Ejemplo 3.1 (Normal de media desconocida)**.** *Consideremos una muestra $\boldsymbol{x}$ de tamaño n distribuida según $\mathcal{N}(\mu_0, \sigma)$ donde σ es conocida. Queremos contrastar $H_0 : \mu_0 = 0$. Tomamos como estadístico test $T = |\overline{X}_m|$ y generamos B muestras $\boldsymbol{x}_1, ..., \boldsymbol{x}_B$ de n variables independientes $\mathcal{N}(0, \sigma)$. Con las B muestras distribuidas según una distribución de probabilidad conocida evaluamos el estadístico $t_b^* = \bar{x}_b$ para $b = 1, ..., B$ y estimamos el p-valor siguiendo el algoritmo anterior.*

3.3. Contrastes de hipótesis nula compuesta

En ocasiones en la distribución nula de T intervendrán parámetros irrelevantes (*nuisance*), parámetros de un modelo estadístico cuya estimación no es el objetivo principal del análisis que se está desarrollando pero que deben ser tenidos en cuenta en la formulación del mismo modelo y llevan a la formulación de hipótesis nulas compuestas.

Al ser la situación más frecuente que la hipótesis nula sea compuesta, el p-valor (3.1) no estará bien definido porque $\mathbb{P}(T \geq t|F)$ dependerá de qué F tomemos ya que no hay una única F que satisfaga H_0. Para solucionar este problema podemos considerar los tres alternativas siguientes:

- Test pivote. Buscamos un estadístico cuya distribución no dependa de los aspectos desconocidos de la distribución de la muestra bajo H_0. Este estadístico T es de tipo pivote, es decir, que su distribución es igual para cualquier F que satisfaga H_0.
- Test condicional. Eliminamos los parámetros desconocidos bajo H_0 condicionando a un estadístico suficiente S. De esta forma convertimos la hipótesis compuesta en simple. El p-valor queda

$$p = \mathbb{P}(T \geq t|S = s, H_0).$$

- Test bootstrap. Estimamos F por una función de distribución $\widehat{F}_0$ que satisfaga H_0 donde los aspectos desconocidos bajo H_0 se reemplazan por estimadores obtenidos a partir de la muestra. Pueden ser estimadores de un parámetro *nuisance* o de la distribución de la muestra $\mathbf{x}$. El p-valor queda

$$p = \mathbb{P}(T \geq t|\widehat{F}_0).$$

En las siguientes secciones analizaremos los casos del test pivote y el test bootstrap. Puede consultarse con mayor detalle el test condicional en [1, sección 1.2.2].

Davison y Hinkley justifican en [6, sección 4.2.5] como el número de muestras afecta a la *potencia del contraste*, la probabilidad de que la hipótesis nula sea rechazada cuando es falsa. Concluye que la pérdida de potencia con $B = 99$ no es importante para $\alpha \geq 0{,}05$ y que, en general, $B = 999$ debería ser un valor seguro para realizar el contraste.

3.3.1. Contrastes de hipótesis basados en un pivote

Como el estadístico test de tipo pivote tiene la misma distribución F_θ para todos los valores $\theta \in \Theta_0$, el algoritmo de la sección anterior será válido para cualquier valor particular de $\theta \in \Theta_0$ como vemos en el siguiente ejemplo:

Ejemplo 3.2 (Igualdad de medias de muestras distribuidas según una normal)**.** *Supongamos que tenemos dos muestras independientes:* $\boldsymbol{x} = (x_1, ..., x_m)$ *distribuida según* $\mathcal{N}(\mu_x, 1)$ *e* $\boldsymbol{y} = (y_1, ..., y_n)$ *distribuida según* $\mathcal{N}(\mu_y, 1)$*. Queremos contrastar* $H_0 : \mu_x = \mu_y$*. En esta situación, la hipótesis nula presenta un parámetro desconocido (*$\mu_x = \mu_y$*).*

Tipificando tenemos que $v_i = x_i - \mu_x, u_i = y_i - \mu_y \sim \mathcal{N}(0,1)$*. Por tanto, el estadístico dado por*

$$t = T(\boldsymbol{x},\boldsymbol{y}) = |\bar{y} - \bar{x}| = \left| \frac{1}{n}\sum_{i=1}^{n} y_i - \frac{1}{m}\sum_{i=1}^{m} x_i \right| = \left| \frac{1}{n}\sum_{i=1}^{n} u_i - \frac{1}{m}\sum_{i=1}^{m} v_i + (\mu_y - \mu_x) \right|, \tag{3.2}$$

es de tipo pivote al no depender de μ_x *y de* μ_y *bajo la hipótesis nula* H_0*. Procedemos como sigue:*

- *Tomamos, por ejemplo,* $\mu_x = \mu_y = 37$*. Notar que es válido tomar cualquier valor de* $\mathbb{R}$*.*
- *Generamos B muestras* $\boldsymbol{x}_1, ..., \boldsymbol{x}_B$*, siendo cada muestra un vector de m valores independientes distribuidos según una normal* $\mathcal{N}(37, 1)$*.*
- *Generamos B muestras* $\boldsymbol{y}_1, ..., \boldsymbol{y}_B$*, siendo cada muestra un vector de n valores independientes distribuidos según una normal* $\mathcal{N}(37, 1)$*.*
- *Evaluamos el estadístico* $t_b^* = T(\boldsymbol{x}_b, \boldsymbol{y}_b)$ *para* $b = 1, ..., B$ *según lo hemos definido en (3.2).*
- *Calculamos* $\hat{p} = \dfrac{1}{B}\displaystyle\sum_{b=1}^{B} \mathbb{1}\{t_b^* \geq t\}$ *y rechazamos* H_0 *si* $\hat{p} \leq \alpha$*.*

3.3.2. Contrastes de hipótesis bootstrap

En muchos problemas, la distribución de T bajo H_0 dependerá de los parámetros *nuisance* que no podrán condicionarse en la distribución, por lo que deberemos buscar una alternativa. En esta situación, la idea del bootstrap es estimar F por $\widehat{F}_0$ y proceder como si esta última fuera la distribución correcta. Es decir, se busca encontrar una aproximación de Monte Carlo al p-valor de la forma

$$p = \mathbb{P}(T(X^*) > t | H_0) \text{ donde } X^* \sim \widehat{F}_0.$$

Al contrario que las secciones previas, esto es solo una aproximación estadística de un p-valor. Pero si H_0 es cierta y $\widehat{F}_0$ es buen estimador, se obtienen resultados válidos como se ve en [1, sección 1.2.3].

La diferencia con los algoritmos bootstrap discutidos anteriormente es que $\widehat{F}_0$ debe satisfacer H_0. Si antes el remuestreo era a partir de la distribución empírica $\widehat{F}$, ahora debemos remuestrear a partir de la distribución $\widehat{F}_0$, que satisface la hipótesis nula H_0.

Ejemplo 3.3 (Contraste de media nula)**.** *Supongamos que tenemos una muestra observada* $\boldsymbol{x} = (x_1, ..., x_n)$ *distribuida según* F_0 *y deseamos contrastar si la media es nula,* $H_0 : E[X_i] = 0$*. En esta situación,* F_0 *no está especificada por la hipótesis nula y podemos estimarla por la función de distribución empírica* $\widehat{F}$*. Sin embargo, si* $Y \sim \widehat{F}$*, entonces* $\mathbb{E}[Y] = \bar{x}$ *que no es necesariamente nula.*

Por tanto, es necesario estimar F_0 *imponiendo la hipótesis nula de que los marginales (cada variable de la muestra) sean de esperanza nula. Para imponer esa condición, trabajamos con la muestra* $u_i = x_i - \bar{x}$ *con* $i = 1, ...n$*, por lo que la función de distribución empírica es* $F_n(u) = \displaystyle\prod_{i=1}^{n} \widehat{F}(x_i - \bar{x})$*, y se verifica que* $\mathbb{E}[U] = \mathbb{E}[X - \mathbb{E}[X]] = 0$*.*

Tomando $T(\boldsymbol{x}) = |\bar{x}|$ *como el estadístico test, procedemos como sigue:*

- *Elegimos enteros* $i_1, ..., i_n$ *que tomen valores entre* 1 *y n con la misma probabilidad* $1/n$*.*
- *Cada muestra bootstrap es* $\boldsymbol{x}_b^* = (x_{i_1} - \bar{x}, ..., x_{i_n} - \bar{x})$*.*
- *Calculamos el estadístico para la muestra bootstrap* $t_b^* = T(\boldsymbol{x}_b^*)$*.*
- *Repetimos los pasos anteriores para* $b = 1, ..., B$*.*
- *Calculamos* $\hat{p} = \dfrac{1}{B}\displaystyle\sum_{b=1}^{B} \mathbb{1}\{t_b^* \geq T(\boldsymbol{x})\}$ *y rechazamos* H_0 *si* $\hat{p} \leq \alpha$*.*

Capítulo 4

Modelos de regresión

Entre los métodos estadísticos más útiles y más utilizados encontramos los modelos de regresión. Permiten análisis relativamente simples de situaciones complicadas, donde tratamos de cuantificar los efectos de muchas variables explicativas o regresoras en una variable respuesta.

En este capítulo comenzamos introduciendo el tratamiento clásico del modelo de regresión lineal mediante métodos de mínimos cuadrados. Posteriormente, veremos dos aplicaciones diferentes del bootstrap en los modelos de regresión basadas en los residuos o en parejas de variables. Para finalizar, veremos intervalos de confianza bootstrap para los parámetros de regresión y analizaremos las dos aplicaciones del bootstrap en regresión para determinar cuál proporciona una mejor información.

4.1. El modelo de regresión lineal clásico y la técnica de mínimos cuadrados

En el **modelo de regresión lineal** tratamos de cuantificar los efectos de *p variables independientes o regresoras*, $X_1,...,X_p$, en Y, la *variable respuesta,* a partir de una muestra de tamaño n de esas variables. Llamaremos *matriz del diseño* a la matriz $\mathbf{X}$ de orden $n \times (p+1)$ donde en la primera columna todos los elementos serán 1 y la $(i+1)$-ésima columna es el vector $\mathbf{x}_i$ para $i = 1,...,p$, es decir, $\mathbf{X} = [\mathbf{1}_n|\mathbf{x}_1|...|\mathbf{x}_p]$. Denotaremos el error del modelo por $\boldsymbol{\varepsilon} = (\varepsilon_1,...,\varepsilon_n)^T$ donde cada *término del error* ε_i es independiente de los demás e independiente de las variables regresoras y del que se asume que tiene distribución Normal, esperanza nula y varianza constante, es decir, $F \to (\varepsilon_1,...,\varepsilon_n) = \boldsymbol{\varepsilon} \quad [E_F[\boldsymbol{\varepsilon}] = 0]$.

Asumiremos que $n >> p$ y que ninguna variable $\mathbf{x}_k$ para $k = 1,...,p$ es combinación lineal del resto de variables regresoras o, de lo contrario, el modelo no sería identificable. De esta forma, la expresión del modelo de regresión lineal con notación matricial es $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, donde $\boldsymbol{\beta} = (\beta_0,...,\beta_p)^T$ es el *vector de parámetros de regresión* desconocido y que deseamos estimar a partir de los datos observados. El *estimador de mínimos cuadrados de* $\boldsymbol{\beta}$ es $\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$ y a partir de las ecuaciones normales de regresión lineal obtenemos que

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}.$$

Notar que la existencia de $(\mathbf{X}^T\mathbf{X})^{-1}$ implica que $n \geq p+1$ y que el rango por columnas de $\mathbf{X}$ es máximo, es decir, $p+1$.

Llamaremos *residuo* a la diferencia entre el valor observado de la variable respuesta y el valor ajustado, es decir, $\mathbf{e} = \mathbf{y} - \hat{\mathbf{y}} = \mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}$.

En las condiciones del modelo clásico de regresión, el Teorema de Gauss-Markov nos garantiza que el estimador de mínimos cuadrados $\hat{\boldsymbol{\beta}}$ es lineal, insesgado de mínima varianza. Bajo estos supuestos, la *matriz de covarianzas* $\boldsymbol{\Sigma}$ del estimador de mínimos cuadrados $\hat{\boldsymbol{\beta}}$ es

$$\Sigma = \mathbb{V}ar(\hat{\boldsymbol{\beta}}) = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbb{V}ar(\mathbf{y})\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1} = \sigma_F^2(\mathbf{X}^T\mathbf{X})^{-1},$$

ya que $\mathbb{V}ar(\mathbf{y}) = \sigma_F^2\mathbb{I}$, donde $\mathbb{I}$ es la matriz identidad y $\sigma_F^2 = \mathbb{V}ar(\boldsymbol{\varepsilon})$.

Las hipótesis del modelo clásico de regresión son bastante restrictivas (normalidad de la respuesta, homocedasticidad, etc.) y si no se verifican estas condiciones no siempre se pueden garantizar todas las buenas propiedades de los estimadores de mínimos cuadrados. Aunque resultados más recientes [14] muestran que estos estimadores mantienen sus buenas propiedades incluso con hipótesis muy generales.

Si los estimadores $\hat{\boldsymbol{\beta}}$ tienen una distribución normal o asintóticamente normal podremos aplicar la teoría estándar. Sin embargo, en otras situaciones no conoceremos la distribución de probabilidad de $\hat{\boldsymbol{\beta}}$. En la próxima sección veremos que el bootstrap nos proporciona un método para aproximar la distribución de $\hat{\boldsymbol{\beta}}$ a través de las muestras bootstrap $\hat{\boldsymbol{\beta}}^*$.

4.2. Aplicación del bootstrap en los modelos de regresión

Existen dos enfoques básicos para aplicar el bootstrap en el problema de regresión. Uno es aplicar el bootstrap a los residuos del modelo que son independientes entre sí y con una misma distribución. El otro es aplicarlo a la pareja formada por la variable respuesta y las variables regresoras.

4.2.1. Bootstrap basado en residuos

Comenzamos aplicando el bootstrap a los residuos del modelo $\mathbf{e} = \mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}$. Para ello, estimaremos la distribución del error, F, por la distribución empírica de los e_i, $\widehat{F}$, que asignará probabilidad $1/n$ a cada e_i y tendrá esperanza nula.

Para aplicar el bootstrap a los residuos seguiremos los siguientes pasos:

- Seleccionamos una muestra aleatoria de residuos bootstrap $\widehat{F} \to (e_1^*, ..., e_n^*)^T = \mathbf{e}^*$ donde cada e_i^* es igual a cualquiera de los valores e_j con probabilidad $1/n$ para $i, j = 1, ..., n$.
- La *variable respuesta bootstrap* $\mathbf{y}^* = (y_1^*, ..., y_n^*)^T$ se obtiene a partir de $y_i^* = \hat{y}_i + e_i^*$.

Con estos cálculos tenemos dos alternativas a seguir:

1. Generar solo una muestra bootstrap. Descrita por Efron y Tibshirani en [12, sección 9.4], tenemos la muestra bootstrap $(x_{i1}, ..., x_{ip}, y_i^*)$ para $i = 1, ..., n$. Obtenemos el *estimador de mínimos cuadrados bootstrap* $\hat{\boldsymbol{\beta}}^* = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}^*$, de donde podemos obtener una expresión matricial para el error estándar bootstrap de las componentes de $\hat{\boldsymbol{\beta}}^*$.

   En efecto, como $\mathbb{V}ar(\mathbf{y}^*) = \widehat{\sigma}_F^2\mathbb{I}$, donde $\mathbb{I}$ es la matriz identidad, tenemos que la estimación bootstrap ideal de la matriz de covarianzas es

   $$\widehat{\mathbb{V}ar}(\hat{\boldsymbol{\beta}}) = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbb{V}ar(\mathbf{y}^*)\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1} = \widehat{\sigma}_F^2(\mathbf{X}^T\mathbf{X})^{-1}.$$

   Notar que esta forma de estimar el error es en el fondo la misma que la tradicional de mínimos cuadrados. En la práctica, estimaremos σ_F^2 por $\widehat{\sigma}_F^2 = \frac{1}{n}\sum_{i=1}^{n} e_i^2$ o por una versión de sesgo corregido a través de la variabilidad no explicada del modelo, obteniendo $\widehat{\sigma}^2 = \sum_{i=1}^{n} \frac{e_i^2}{n-p}$ con p el número de variables regresoras en el modelo. Notar que difieren ligeramente y, para valores grandes de n, la diferencia es irrelevante.

2. Como puede verse en [8, sección 5.7] y [5, sección 2.4.2], tomamos la estimación $\hat{\boldsymbol{\beta}}^* = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}^*$ y repetimos el proceso B veces, obteniendo las estimaciones bootstrap $\hat{\boldsymbol{\beta}}_1^*, ..., \hat{\boldsymbol{\beta}}_B^*$.

Definimos $\overline{\hat{\beta}^*} = \frac{1}{B}\sum_{j=1}^{B} \hat{\boldsymbol{\beta}}^*_j$ y tenemos que

$$\widehat{\mathbb{V}ar}(\hat{\boldsymbol{\beta}}) = \frac{1}{B-1}\sum_{j=1}^{B}(\hat{\boldsymbol{\beta}}^*_j - \overline{\hat{\beta}^*})(\hat{\boldsymbol{\beta}}^*_j - \overline{\hat{\beta}^*})^T.$$

4.2.2. Bootstrap basado en parejas

La segunda opción es aplicar el método bootstrap al vector de dimensión $p+1$, $(x_{i1},...,x_{ip},y_i)$ para $i=1,...,n$, formado por la i-ésima componente de las p variables regresoras y de la variable respuesta. El bootstrap aplicado a la pareja trata este vector como un vector aleatorio independiente e idénticamente distribuido con una función de distribución F. Bajo estas suposiciones, será fácil muestrear los vectores con reemplazamiento de la misma forma que cuando procedemos para generar las muestras bootstrap como vemos a continuación:

- Para $i=1,...,n$, obtenemos la muestra $(x^*_{i1},...,x^*_{ip},y^*_i)$ a partir del remuestreo con reemplazamiento del conjunto $\left\{(x_{j1},...,x_{jp},y_j)\right\}_{j=1}^{n}$.
- Definimos la muestra $\mathbf{y}^* = (y^*_1,...,y^*_n)^T$ y construimos la matriz

$$\mathbf{X}_* = \begin{bmatrix} 1 & x^*_{11} & \cdots & x^*_{1p} \\ 1 & x^*_{21} & \cdots & x^*_{2p} \\ \vdots & \vdots & & \vdots \\ 1 & x^*_{p1} & \cdots & x^*_{pp} \end{bmatrix}.$$

- Obtenemos el estimador de mínimos cuadrados bootstrap $\hat{\boldsymbol{\beta}}^* = (\mathbf{X}_*^T\mathbf{X}_*)^{-1}\mathbf{X}_*^T\mathbf{y}^*$.
- Repetimos B veces los pasos anteriores para obtener la distribución bootstrap de $\hat{\boldsymbol{\beta}}^*$ como en la segunda parte del caso anterior.

4.2.3. Análisis de los dos métodos bootstrap

Chernick en [4, sección 4.0] recoge el debate que existe en torno a la aplicación de métodos bootstrap a los residuos o a la pareja de variables. Algunos matemáticos consideran el segundo método como un enfoque inapropiado ya que el problema de regresión requiere que las variables regresoras estén fijadas en el modelo y no seleccionadas al azar de una distribución de probabilidad. La aplicación del bootstrap a la pareja de variables asume implícitamente una distribución de probabilidad conjunta para el vector.

Sin embargo, desde el punto de vista práctico, si este segundo método tiene buenas propiedades de robustez en relación con la especificación del modelo, está justificado su uso tal y como sugieren Efron y Tibshirani en [12, sección 9.5] para el caso de una sola variable regresora. Afirman que los dos enfoques son asintóticamente equivalentes pero que pueden funcionar de manera bastante diferente en muestras pequeñas. También que el bootstrap aplicado a la pareja es menos sensible a desviaciones de las hipótesis del modelo y que, además, solo aplicando el bootstrap a los residuos podemos obtener la estimación bootstrap ideal de la matriz de covarianzas.

Chernick en [4, sección 4.1.3] destaca la importancia de analizar el comportamiento del bootstrap cuando no disponemos de una teoría adecuada para la estimación de la matriz de covarianzas. Especialmente destaca como situaciones a considerar la presencia de heterocedasticidad en la varianza residual, correlación en los residuos, modelos no lineales o que los errores no se distribuyan de forma gaussiana. Chernick afirma que el bootstrap aplicado a la pareja proporciona mejores resultados cuando haya heterocedasticidad en la varianza residual, correlación en los residuos o sospechemos que puedan faltar otros parámetros importantes en el modelo.

4.2.4. Intervalos de confianza bootstrap de los parámetros de regresión

En el Capítulo anterior hemos visto dos métodos bootstrap para construir intervalos de confianza. En esta sección recordamos el procedimiento para la obtención de los intervalos bootstrap-*t* y de tipo percentil adaptado a la notación de los modelos de regresión y de los que realizaremos un análisis comparativo basado en simulaciones en el siguiente Capítulo.

También es importante señalar que dichos procedimientos se pueden aplicar a cualquiera de los dos métodos que hemos visto en las secciones anteriores para obtener muestras de las estimaciones bootstrap $\hat{\beta}_1^*, ...\hat{\beta}_B^*$, el bootstrap basado en residuos y en parejas.

Intervalo bootstrap-t

Seguimos el procedimiento visto en la sección [2.2]:

- Por cada muestra y componente calculamos $Z_{b,j}^* = \dfrac{\hat{\beta}_{b,j}^* - \hat{\beta}_j}{\widehat{se}_b^*(\hat{\beta}_{b,j}^*)}$ donde $\hat{\beta}_{b,j}^*$ es la componente j-ésima de $\hat{\boldsymbol{\beta}}_b^*$ para $b = 1, ..., B$ y $j = 1, ..., n$.
- El α-ésimo percentil de $Z_{b,j}^*$ es estimado por el valor $\hat{t}_\alpha$ tal que $\#\left\{Z_{b,j}^* \leq \hat{t}_\alpha\right\}/B = \alpha$.
- El *intervalo bootstrap-t* de β_j para $j = 1, ..., n$ es

$$\left[\hat{\beta}_j - \hat{t}_{1-\alpha} \cdot \widehat{se}(\hat{\beta}_j), \hat{\beta}_j - \hat{t}_\alpha \cdot \widehat{se}(\hat{\beta}_j)\right].$$

Intervalo bootstrap de tipo percentil

Seguimos el procedimiento visto en la sección [2.3]:

- Denotamos por $\hat{\beta}_{B(\alpha),j}^*$ el percentil $100 \cdot \alpha$-ésimo de los valores $\hat{\beta}_{b,j}^*$, es decir, el $B \cdot \alpha$-ésimo valor en la muestra ordenada de las j-ésimas componentes de $\hat{\boldsymbol{\beta}}_1^*, ..., \hat{\boldsymbol{\beta}}_B^*$ con $j = 1, ..., n$.
- Análogamente obtenemos que $\hat{\beta}_{B(1-\alpha),j}^*$ es el percentil $100 \cdot (1-\alpha)$ de la muestra ordenada de las j-ésimas componentes.
- El *intervalo bootstrap de tipo percentil* $1 - 2\alpha$ de β_j con $j = 1, ..., n$ es

$$\left[\hat{\beta}_{B(\alpha),j}^*, \hat{\beta}_{B(1-\alpha),j}^*\right].$$

Capítulo 5

Análisis comparativo basado en simulaciones de la precisión de los intervalos de confianza de los coeficientes de regresión

El objetivo de este capítulo es realizar un estudio de simulación para **analizar el comportamiento** de los intervalos de confianza de los estimadores de los coeficientes de regresión, $\hat{\beta}_j$, obtenidos con los distintos métodos considerados en el capítulo anterior. En particular, vamos a analizar el nivel de confianza o cubrimiento y la amplitud de estos intervalos de confianza bajo distintas situaciones (consideraremos diferentes distribuciones del error $\boldsymbol{\varepsilon}$ y distintos tamaños de muestra). De esta forma, se trata de obtener evidencias de que método puede ser más adecuado en cada situación.

5.1. Idea intuitiva del algoritmo

Para realizar nuestro análisis creamos un algoritmo del que, a continuación, resumimos las ideas principales para mostrar de forma intuitiva su funcionamiento. Puede consultarse en detalle su código R en la sección [A.1] del Apéndice.

- **Realizaremos simulaciones** de un modelo de regresión lineal dado por $Y = \beta_0 + \beta_1 X + \boldsymbol{\varepsilon}$ donde $\beta_0 = 1{,}5$ y $\beta_1 = 2{,}1$. Consideraremos en primer lugar $\boldsymbol{\varepsilon} \sim \mathcal{N}(0,1)$ y, en segundo lugar, consideraremos $\boldsymbol{\varepsilon} \sim \mathcal{E}(1)$. Así mismo, consideraremos diferentes **tamaños de muestra** de 25 y 250 para la variable regresora. Notar que una vez que generamos la muestra inicial **x**, permanece fija a lo largo de la simulación pues consideramos que **x** es determinista, es decir, no aleatoria; se ha utilizado $X \sim \mathcal{U}(0,1)$.

- Nos interesamos por la calidad de las estimaciones de β_0 y β_1 y su variabilidad. Para cada coeficiente del modelo β_i, **calcularemos su intervalo de confianza** basado en la teoría clásica de mínimos cuadrados y los intervalos de confianza bootstrap-t y de tipo percentil resultado de aplicar el bootstrap basado en parejas y en los residuos.

- Para cada intervalo obtenido, **comprobamos si contiene el valor real** del parámetro β_i y **calculamos su amplitud**.

- **Repetimos los pasos anteriores** 1000 veces.

- Por cada tipo de intervalo, **estimamos el nivel de confianza** como el porcentaje de intervalos de confianza que contienen el valor real del parámetro y tomamos el valor medio de la amplitud del intervalo dada por la diferencia de sus extremos.

5.2. Resultados obtenidos e interpretación

Por cada iteración del algoritmo obtenemos, como resultado de aplicar mínimos cuadrados o bootstrap basado en la pareja o residuos, diferentes intervalos de confianza, sus amplitudes y si el valor real de los coeficientes β_i está contenido en cada intervalo respectivo. A continuación, mostramos los resultados obtenidos tras realizar 1000 simulaciones para las diferentes situaciones que estamos considerando respecto la distribución del error y el tamaño muestral.

β_0			Mínimos cuadrados	Intervalo bootstrap-t		Intervalo tipo percentil	
				Pareja	Residuos	Pareja	Residuos
$\varepsilon \sim \mathcal{N}(0,1)$	$n=25$	Nivel confianza	0,943	0,933	0,941	0,925	0,916
		Amplitud	1,649	1,643	1,656	1,537	1,499
	$n=250$	Nivel confianza	0,950	0,952	0,947	0,950	0,947
		Amplitud	0,497	0,496	0,497	0,493	0,493
$\varepsilon \sim \mathcal{E}(1)$	$n=25$	Nivel confianza	0,933	0,885	0,937	0,878	0,910
		Amplitud	1,588	1,689	1,634	1,455	1,435
	$n=250$	Nivel confianza	0,950	0,948	0,954	0,950	0,949
		Amplitud	0,507	0,508	0,510	0,503	0,504
β_1			Mínimos cuadrados	Intervalo bootstrap-t		Intervalo tipo percentil	
				Pareja	Residuos	Pareja	Residuos
$\varepsilon \sim \mathcal{N}(0,1)$	$n=25$	Nivel confianza	0,952	0,934	0,947	0,926	0,917
		Amplitud	3,131	3,130	3,140	2,938	2,852
	$n=250$	Nivel confianza	0,950	0,948	0,948	0,941	0,945
		Amplitud	0,891	0,889	0,891	0,885	0,884
$\varepsilon \sim \mathcal{E}(1)$	$n=25$	Nivel confianza	0,942	0,926	0,940	0,910	0,925
		Amplitud	2,940	3,132	2,915	2,742	2,697
	$n=250$	Nivel confianza	0,944	0,940	0,946	0,940	0,945
		Amplitud	0,908	0,907	0,910	0,902	0,903

Tabla 5.1: Estimación del nivel de confianza y valor medio de la amplitud de los intervalos de confianza de β_0 y β_1 obtenidos tras 1000 simulaciones aplicando mínimos cuadrados y procedimientos bootstrap basados en la pareja y en los residuos.

De la observación de la tabla [5.1] obtenemos las siguientes conclusiones tanto para β_0 como para β_1:

- Cuando la distribución es normal y el tamaño de muestra pequeño, vemos que el intervalo obtenido por mínimos cuadrados y el bootstrap-t resultado de aplicar bootstrap basado en residuos han proporcionado un mejor nivel de confianza alcanzando el 95 % para el β_1. La amplitud de los intervalos de confianza es más pequeña para el intervalo percentil basado en los residuos, si bien es el que presenta un menor nivel de confianza.
- Cuando la distribución es normal y el tamaño muestral grande, el nivel de confianza de todos los intervalos es del 95 %, salvo en el caso del intervalo de tipo percentil aplicado a la pareja que es un punto inferior para el β_1. La amplitud es idéntica en todos los casos con diferencias de una centésima.
- Cuando la distribución es exponencial y el tamaño muestral pequeño, vemos de nuevo que el intervalo de confianza obtenido por mínimos cuadrados y el intervalo bootstrap-t basado en los residuos han proporcionado un mejor nivel de confianza sin alcanzar el 95 % en ninguno de los casos. El intervalo de tipo percentil basado en los residuos también presenta una amplitud más pequeña.

- Cuando la distribución es exponencial y el tamaño de muestra grande, el nivel de confianza de los intervalos es próximo al valor deseado. Solo se alcanza un nivel de confianza del 95 % en todos los casos para el β_0 y en los que resultan al aplicar métodos bootstrap basados en residuos para el β_1. La amplitud es idéntica en todos los casos con diferencias de una centésima.

En general, en todas las situaciones se observa que cuanto mayor es el nivel de confianza alcanzado, más amplio es el intervalo. Cuando la distribución del error es normal, como es de esperar obtenemos buenos resultados cuando el tamaño de la muestra es grande por el comportamiento asintótico de la distribución. Entre los diferentes intervalos bootstrap, el intervalo bootstrap-*t* obtenido al aplicar métodos bootstrap basados en los residuos nos proporciona resultados iguales o próximos al valor de significación que deseamos en todos los casos.

Respecto al nivel de confianza los intervalos bootstrap-*t* ofrecen un nivel más cercano al 95 % que los de tipo percentil, especialmente cuando el tamaño de la muestra es pequeño. Así mismo, el bootstrap basado en residuos obtiene un nivel más próximo al 95 % que el basado en parejas cuando la muestra es pequeña. Cuando el tamaño dela muestra es grande, todos los métodos obtienen resultados similares.

En relación a la amplitud de los intervalos de confianza observamos que los intervalos de tipo percentil tienen una amplitud inferior que los intervalos bootstrap-*t* en todos los casos. Y lo mismo sucede si comparamos los obtenidos al aplicar el bootstrap basado en residuos frente al basado en parejas.

En [2] se prueba que si las simulaciones se realizan de forma consistente con el modelo, el bootstrap proporcionará los mismos resultados asintóticos que los métodos clásicos. En [13] se analiza con detalle que para obtener intervalos de confianza con una mayor precisión que los intervalos *t* y bootstrap-*t*, es preciso realizar correcciones, entre otras, en la asimetría y la curtosis de la distribución del estadístico.

Bibliografía

[1] BAXEVANI, A., (2011). *"MSA100, Computer Intensive Statistical Methods"*, Mathematical Sciences - Chalmers University of Technology and University of Gothenburg. http://www.math.chalmers.se/Stat/Grundutb/GU/MSA100/A11/lecture7.pdf.

[2] BOIK, R.J., (2008). *Accurate confidence intervals in regression analyses of non-normal data*, Annals of the Institute of Statistical Mathematics, **60**(1), 61-83. https://doi.org/10.1007/s10463-006-0085-1.

[3] CANTY, A. & RIPLEY, B., *boot: Bootstrap Functions (Originally by Angelo Canty for S)* (Version 1.3-22). https://cran.r-project.org/package=boot.

[4] CHERNICK, M. R., (2007). *Bootstrap Methods: A Guide for Practitioners and Researchers*, John Wiley & Sons.

[5] CHERNICK, M. R., & LABUDDE, R.A., (2011). *An Introduction to Bootstrap Methods with Applications to R*, John Wiley & Sons.

[6] DAVISON, A. C., & HINKLEY, D. V., (1997). *Bootstrap Methods and their Application*, Cambridge University Press.

[7] EFRON, B., (1979). *Bootstrap Methods: Another Look at the Jackknife*, The Annals of Statistics **7**(1), 1-26. https://www.jstor.org/stable/2958830.

[8] EFRON, B., (1982). *The Jackknife, the Bootstrap, and Other Resampling Plans*, CBMS-NSF Regional Conference Series in Applied Mathematics **38**, SIAM.

[9] EFRON, B., & TIBSHIRANI, R.J., (1986). *Bootstrap methods for standard errors, confidence intervals and other measures of statistical accuracy*, Statical Science **1**, 54-77. https://www.jstor.org/stable/2245500.

[10] EFRON, B., (1987). *Better bootstrap confidence intervals*, Journal of the American Statistical Association **82**(397), 171-200.

[11] EFRON, B., (1988). *Bootstrap confidence intervals. Good or bad?*, Psychological Bulletin **104**(2), 293-296. http://dx.doi.org/10.1037/0033-2909.104.2.293.

[12] EFRON, B., & TIBSHIRANI, R.J., (1993). *An Introduction to the Bootstrap*, Monographs on Statistics and Applied Probability 57, Chapman & Hall, New York.

[13] FREEDMAN, D. A., (1981). *Bootstrapping Regression Models*, The Annals of Statistics **9**(6), 1218-1228. https://www.jstor.org/stable/2240411.

[14] HAYASHI, F., (2000). *Econometrics*, Princeton: Princeton University Press.

[15] NOREEN, E. W., (1989). *Computer Intensive Methods for Testing Hypotheses: An Introduction*, John Wiley & Sons.

[16] PENG, R.D., *simpleboot: Simple Bootstrap Routines* (Version 1.1-7). `https://cran.r-project.org/package=simpleboot`.

[17] TIBSHIRANI, R., *bootstrap: Functions for the Book "An Introduction to the Bootstrap"* (Version 2017.2). `https://cran.r-project.org/package=bootstrap`.

Apéndice A

Ficheros de código R

En este apéndice se recogen los ficheros de código R de los que se hace referencia en algunas secciones del presente trabajo. Se presentan en primer lugar los referentes al Capítulo 5 pues en ellos se incluyen comentarios que detallan las diferentes funciones utilizadas del paquete `boot`, así como parte de la documentación incluida en este, que facilitan la comprensión del código. Por tanto, es recomendable su primera lectura para un mejor seguimiento de los demás.

El paquete `boot` implementa las funciones y conjuntos de datos del libro de Davison y Hinkley [6, Capítulo 11]. También existen otros paquetes como `simpleboot` y `bootstrap` que recoge las funciones y datos del libro de Efron y Tibshirani [12, Apéndice]. Utilizaremos el primero de ellos ya que presenta una mayor variedad de funciones y resultados. Para mayor información y detalle, puede consultarse la documentación de los paquetes respectivamente en [3], [16] y [17].

A.1. Capítulo 5: Análisis comparativo basado en simulaciones de la precisión de los intervalos de confianza de los coeficientes de regresión

Para facilitar la comprensión de los resultados obtenidos comenzamos presentando el algoritmo base para una única simulación que reproduce los tres primeros pasos expuestos en la sección [5.1]. Posteriormente se recoge el algoritmo completo con las modificaciones necesarias para realizar 1000 simulaciones.

A.1.1. Código para una única simulación

```
##PARÁMETROS DEL MODELO##
#Fijamos los valores de los parámetros del modelo
b0=1.5
b1=2.1
sigma=1 #varianza de la normal o parámetro de la exponencial
nreplicas = 2000 #número de réplicas bootstrap

#Fijamos los valores de la variable independiente
nsample = 250 #tamaño de la muestra
x = runif(nsample, 0,1) #distribución de las variables regresoras

#Simulamos un modelo de regresión lineal con errores distribuidos según
  #una normal de media 0 y varianza sigma o
  #una exponencial de parámetro sigma.
#En el segundo caso restamos 1/sigma para generar un error con distribución
#exponencial de parámetro 1 desplazada de forma que la media del error es
#cero y así evitamos que se sume a la estimación del beta0.
```

```
#Comentar una de las dos opciones según el modelo que deseemos simular
y = b0 + b1*x + rnorm(nsample,0,sigma)
#y = b0 + b1*x + rexp(nsample,sigma)-1/sigma

##RESULTADOS CLÁSICOS. MÍNIMOS CUADRADOS##
lmodel = lm(y ~ x)
summary(lmodel)

datos = data.frame (y,x)

#Coeficientes del modelo
beta = coef(lmodel)
beta

#Intervalos de confianza para los coeficientes
confint(lmodel)

##RESULTADOS BOOTSTRAP##
#Utilizamos el paquete boot
library (boot)

##REGRESIÓN BOOTSTRAP BASADA EN PAREJAS O FILAS##
#Creamos una función que será el estadístico de interés que nos hará falta
#después para generar las réplicas bootstrap.
  #Se generan n valores comprendidos entre 1 y n pudiéndose repetir, se crea el
  #nuevo conjunto de datos tomando el i-ésimo valor del conjunto original según
  #los valores generados.
  #Se realiza el modelo de regresión lineal con el nuevo conjunto y se devuelven
  #los coeficientes del modelo y la diagonal de la matriz de
  #varianzas-covarianzas que son las varianzas de las estimaciones de los
  #parámetros del modelo.
boot.pair = function (datos,i){
  model = lm(y ~ x, data=datos[i,])
  c(coef(model),diag(vcov(model)))
}

#Generación de las réplicas bootstrap del estadístico
##boot##
  #Generar R réplicas bootstrap de un estadístico aplicado a los datos.
  #Tanto remuestreo paramétrico como no paramétrico son posibles.
  #boot(data, statistic, R, sim = "ordinary", stype = c("i", "f", "w"),
  #    strata = rep(1,n), L = NULL, m = 0, weights = NULL,
  #    ran.gen = function(d, p) d, mle = NULL, simple = FALSE, ...,
  #    parallel = c("no", "multicore", "snow"),
  #    ncpus = getOption("boot.ncpus", 1L), cl = NULL)
  #data -> Los datos como vector, matriz o frame de datos.
  #statistic -> Función que, cuando se aplica a los datos, devuelve un vector que
  #             contiene los estadísticos de interés. El estadístico debe tomar
  #             al menos dos argumentos. El primer argumento pasado siempre serán
  #             los datos originales. El segundo será un vector de índices,
```

```
  #             frecuencias o pesos que definan la muestra bootstrap.
  #R -> El número de réplicas bootstrap.
boot1 = boot(data=datos, statistic = boot.pair, R=nreplicas)

#Valor puntual de las estimaciones bootstrap dado por la media de todas ellas
mean(boot1$t[,1])
mean(boot1$t[,2])

##print##
  #Método para la función print() para objetos de la clase "boot".
  #Para cada estadístico calculado mediante bootstrap se muestra el valor original
  #y las estimaciones bootstrap del sesgo y el error estándar.
  #print(x, digits = getOption("digits"), index = 1:ncol(boot.out$t), .)
  #x -> Objeto bootstrap de la clase "boot".
  #digits -> Número de dígitos a mostrar en las estadísticas del resumen.
  #index -> Los índices indican qué elementos de las estadísticas del resumen son
  #         necesarios en la salida.
print(boot1)

#Intervalos de confianza bootstrap-t y de percentiles
##boot.ci##
  #Genera cinco tipos de intervalos de confianza no paramétricos.
  #Utilizaremos solo dos, el bootstrap-t o studentizado y el de percentiles.
  #boot.ci(boot.out, conf = 0.95, type = c("norm","basic", "stud", "perc", "bca"),
  #      index = 1:min(2,length(boot.out$t0)), var.t0 = NULL,
  #      var.t = NULL, t0 = NULL, t = NULL, L = NULL,
  #      h = function(t) t, hdot = function(t) rep(1,length(t)),
  #      hinv = function(t) t, .)
  #boot.out -> Objeto bootstrap de la clase "boot".
  #conf -> Escalar o vector que contenga los niveles de confianza requeridos.
  #type -> Vector de cadenas de caracteres representando los tipos de intervalos
  #        requeridos.
  #index -> Vector de longitud 1 o 2. El primer elemento indica la posición de
  #         la variable de interés en boot.out. El segundo elemento indica la
  #         posición de la varianza de la variable de interés.
boot.ci(boot1, conf = 0.95, type = c("stud","perc"), index = c(1,3)) #IC para beta0
boot.ci(boot1, conf = 0.95, type = c("stud","perc"), index = c(2,4)) #IC para beta1

#Representación de las réplicas bootstrap
##plot##
  #Toma un objeto bootstrap y produce gráficos para las réplicas bootstrap de la
  #variable de interés. Generalmente producirá dos gráficos. El gráfico de la
  #izquierda será un histograma de las réplicas bootstrap. Una línea discontinua
  #vertical indica la posición del valor real del estadístico. El gráfico de la
  #derecha es un Q-Q plot de las réplicas bootstrap.
  #plot(x, index = 1, t0 = NULL, t = NULL, jack = FALSE, qdist = "norm",
  #    nclass = NULL, df, .)
  #x -> Objeto bootstrap de la case "boot".
  #index -> Índice de la variable de interés dentro de la salida de boot.out.
plot(boot1,index=1)
plot(boot1,index=2)
```

```
##REGRESIÓN BOOTSTRAP BASADA EN RESIDUOS##
#Análogamente al bootstrap basado en parejas generamos una función que devuelva
#nuestro estadístico de interés, realizaremos las réplicas bootstrap y obtendremos
#las estimaciones puntuales y los intervalos de confianza bootstrap-t y de
#percentiles.
boot.res = function (datos,i){
  model = lm(y ~ x, data = datos)
  yhat = fitted(model)
  e = resid(model)
  y.star = yhat + e[i]
  model2 = lm(y.star ~ x)
  c(coef(model2),diag(vcov(model2)))
}

boot2 = boot(data=datos, statistic=boot.res, R=nreplicas)

mean(boot2$t[,1])
mean(boot2$t[,2])

boot2

boot.ci(boot2, conf = 0.95, type = c("stud","perc"), index = c(1,3))
boot.ci(boot2, conf = 0.95, type = c("stud","perc"), index = c(2,4))

plot(boot2, index=1)
plot(boot2, index=2)
```

	$\mathcal{N}(0,1), n=25$		$\mathcal{N}(0,1), n=250$	
	β_0	β_1	β_0	β_1
Valor del parámetro	1.5	2.1	1.5	2.1
	Mínimos cuadrados			
Estimación del parámetro	1.392	2.815	1.338	2.345
Intervalo de confianza	[0.581,2.202]	[1.253,4.377]	[1.085,1.591]	[1.907,2.783]
Error estándar	0.392	0.755	0.129	0.222
	Bootstrap basado en la pareja			
Estimación del parámetro	1.408	2.800	1.340	2.342
Intervalo de confianza bootstrap-t	[0.643,2.020]	[1.511,4.341]	[1.094,1.585]	[1.933,2.751]
Intervalo de confianza de percentiles	[0.742,2.034]	[1.419,4.216]	[1.088,1.588]	[1.932,2.749]
Error estándar	0.330	0.705	0.127	0.208
	Bootstrap basado en los residuos			
Estimación del parámetro	1.391	2.820	1.339	2.345
Intervalo de confianza bootstrap-t	[0.604,2.173]	[1.266,4.396]	[1.079,1.594]	[1.898,2.793]
Intervalo de confianza de percentiles	[0.658,2.078]	[1.372,4.245]	[1.084,1.600]	[1.902,2.446]
Error estándar	0.371	0.723	0.129	0.220
	$\mathcal{E}(1), n=25$		$\mathcal{E}(1), n=250$	
	β_0	β_1	β_0	β_1
Valor del parámetro	1.5	2.1	1.5	2.1
	Mínimos cuadrados			
Estimación del parámetro	1.404	2.273	1.404	2.023
Intervalo de confianza	[0.613,2.196]	[0.560,3.985]	[1.174,1.634]	[1.614,2.432]
Error estándar	0.383	0.828	0.117	0.208
	Bootstrap basado en la pareja			
Estimación del parámetro	1.412	2.227	1.401	2.025
Intervalo de confianza bootstrap-t	[0.736,2.605]	[0.083,3.751]	[1.164,1.650]	[1.566,2.485]
Intervalo de confianza de percentiles	[0.728,2.280]	[0.643,3.612]	[1.176,1.657]	[1.572,2.471]
Error estándar	0.396	0.767	0.125	0.228
	Bootstrap basado en los residuos			
Estimación del parámetro	1.420	2.255	1.404	2.021
Intervalo de confianza bootstrap-t	[0.707,2.406]	[0.425,3.822]	[1.190,1.638]	[1.607,2.433]
Intervalo de confianza de percentiles	[0.764,2.207]	[0.811,3.884]	[1.187,1.639]	[1.617,2.441]
Error estándar	0.372	0.793	0.115	0.207

Tabla A.1: Resultados de una simulación con $\beta_0 = 1{,}5$ y $\beta_1 = 2{,}1$

A.1.2. Código para varias simulaciones

El siguiente código es utilizado para obtener los resultados expuestos en la sección [5.2]. En ella, deseamos estudiar el nivel de confianza de los intervalos obtenidos, que definimos en el 95 % y su amplitud. Para ello, implementamos en el código anterior un control que verifique si los intervalos abarcan los valores reales de los parámetros. Para guardar la información, creamos dos matrices de 10 columnas y tantas filas como número de veces repitamos la simulación (en nuestro caso 1000) donde, cada componente de las columnas impares tomará el valor 1 o 0 dependiendo de si los valores reales de los parámetros del modelo, $\beta_0 = 1{,}5$ y $\beta_1 = 2{,}1$, se encuentren o no en los intervalos de confianza respectivos y guardaremos en las columnas pares la amplitud del intervalo respectivo. La información exacta que se guarda en cada componente, se encuentra comentada en el código.

Finalmente, una vez ejecutadas todas las simulaciones, creamos una matriz de tamaño 2×10 donde guardaremos la frecuencia con la que se encuentran en los intervalos de confianza los valores originales de β_0 y β_1 respectivamente y sus amplitudes medias.

```
##PARÁMETROS DEL MODELO##
#Fijamos los valores de los parámetros del modelo
b0=1.5
b1=2.1
sigma=1 #varianza de la normal o parámetro de la exponencial
nreplicas = 2000 #número de réplicas bootstrap

#Fijamos los valores de la variable independiente
nsample = 250 #tamaño de la muestra
x = runif(nsample, 0,1) #distribución de las variables regresoras

##HERRAMIENTAS PARA LA VERIFICACIÓN DE LA PERTENENCIA AL INTERVALO DE CONFIANZA##
#Creamos una función que devuelve 0 si un valor dado no está en IC dado y 1 si está.
#Dado el valor del parámetro y los extremos del intervalo, el valor estará
#contenido en el intervalo si es mayor que el extremo inferior y menor que
#el extremo superior.
modmatriz = function (valbeta,IC0,IC1){
  if (IC0<=valbeta & valbeta<=IC1) 1
  else 0
}

#Creamos las matrices que almacenan información sobre pertenencia al IC.
#En A y B guardaremos los 1 y 0 sobre pertenencia de beta0 y beta1
#en cada IC para cada simulación y la amplitud del IC.
nsim=1000 #número de simulaciones que realizamos
A=matrix(nrow=nsim, ncol=10)
B=matrix(nrow=nsim, ncol=10)

##FUNCIONES DE LOS ESTADÍSTICOS UTILIZADOS EN LAS RÉPLICAS BOOTSTRAP##
boot.pair = function (datos,i){
  model = lm(y ~ x, data=datos[i,])
  c(coef(model),diag(vcov(model)))
}
```

```
boot.res = function (datos,i){
  model = lm(y ~ x, data = datos)
  yhat = fitted(model)
  e = resid(model)
  y.star = yhat + e[i]
  model2 = lm(y.star ~ x)
  c(coef(model2),diag(vcov(model2)))
}

##EJECUCIÓN DE LAS nsim SIMULACIONES##
#Implementamos el for para realizar las nsim simulaciones.
#Dentro del for se encuentra el mismo código que la sección anterior con las
#modificaciones necesarias para almacenar la información en las matrices creadas.

for(i in 1:nsim){
print(i) #Muestra en pantalla el número de la simulación que se está realizando

#Simulamos un modelo de regresión lineal con errores distribuidos según
  #una normal de media 0 y varianza sigma o
  #una exponencial de parámetro sigma.

#DESCOMENTAR la opción según el modelo que deseemos simular
y = b0 + b1*x + rnorm(nsample,0,sigma)
#y = b0 + b1*x + rexp(nsample,sigma)-1/sigma

##RESULTADOS CLÁSICOS. MÍNIMOS CUADRADOS##
lmodel = lm(y ~ x)

datos = data.frame (y,x)

#Coeficientes del modelo de mínimos cuadrados.
beta = coef(lmodel)

#Intervalo de confianza estándar. Almacenamiento de la permanencia de beta0
#o beta1 en la columna primera de las matrices correspondientes y de la amplitud
#en la columna sexta.
ICnor=confint(lmodel)

A[i,1]=modmatriz(b0,ICnor[1,1],ICnor[1,2])
B[i,1]=modmatriz(b1,ICnor[2,1],ICnor[2,2])
A[i,6]=ICnor[1,2]-ICnor[1,1]
B[i,6]=ICnor[2,2]-ICnor[2,1]

##RESULTADOS BOOTSTRAP##
library (boot)

##REGRESIÓN BOOTSTRAP BASADA EN PAREJAS O FILAS##
boot1 = boot(data=datos, statistic = boot.pair, R=nreplicas)
```

```
#Intervalos de confianza bootstrap-t y de percentiles y almacenamiento de la
#permanencia de beta0 o beta1 en las columnas segunda y tercera de las
#matrices correspondientes y amplitudes de los IC
#en columnas séptima y octava.
ICbootpair0=boot.ci(boot1, conf = 0.95, type = c("stud","perc"), index = c(1,3))
ICbootpair1=boot.ci(boot1, conf = 0.95, type = c("stud","perc"), index = c(2,4))

A[i,2]=modmatriz(b0,ICbootpair0$stud[4],ICbootpair0$stud[5])
B[i,2]=modmatriz(b1,ICbootpair1$stud[4],ICbootpair1$stud[5])

A[i,7]=ICbootpair0$stud[5]-ICbootpair0$stud[4]
B[i,7]=ICbootpair1$stud[5]-ICbootpair1$stud[4]

A[i,3]=modmatriz(b0,ICbootpair0$perc[4],ICbootpair0$perc[5])
B[i,3]=modmatriz(b1,ICbootpair1$perc[4],ICbootpair1$perc[5])

A[i,8]=ICbootpair0$perc[5]-ICbootpair0$perc[4]
B[i,8]=ICbootpair1$perc[5]-ICbootpair1$perc[4]

##REGRESIÓN BOOTSTRAP BASADA EN RESIDUOS##
boot2 = boot(data=datos, statistic=boot.res, R=nreplicas)

#Intervalos de confianza bootstrap-t y de percentiles y almacenamiento de la
#permanencia de beta0 o beta1 en las columnas cuarta y quinta de las
#matrices correspondientes y amplitudes de los IC
#en columnas novena y décima.
ICbootres0=boot.ci(boot2, conf = 0.95, type = c("stud","perc"), index = c(1,3))
ICbootres1=boot.ci(boot2, conf = 0.95, type = c("stud","perc"), index = c(2,4))

A[i,4]=modmatriz(b0,ICbootres0$stud[4],ICbootres0$stud[5])
B[i,4]=modmatriz(b1,ICbootres1$stud[4],ICbootres1$stud[5])

A[i,9]=ICbootres0$stud[5]-ICbootres0$stud[4]
B[i,9]=ICbootres1$stud[5]-ICbootres1$stud[4]

A[i,5]=modmatriz(b0,ICbootres0$perc[4],ICbootres0$perc[5])
B[i,5]=modmatriz(b1,ICbootres1$perc[4],ICbootres1$perc[5])

A[i,10]=ICbootres0$perc[5]-ICbootres0$perc[4]
B[i,10]=ICbootres1$perc[5]-ICbootres1$perc[4]
}


##ESTIMACIÓN DEL NIVEL DE CONFIANZA Y AMPLITUD MEDIA##
#Creamos la matriz donde guardaremos la estimación del nivel de confianza de
#cada IC y la media de sus amplitudes.
#En el siguiente for sumamos los elementos por columnas
#y estimamos el nivel de confianza y la amplitud media.

C=matrix(nrow=2,ncol=10)
k=0
```

```
for(j in 1:5){
  n0=0
  n1=0
  n2=0
  n3=0
  for(i in 1:nsim){
    n0=n0+A[i,j]
    n1=n1+B[i,j]
    n2=n2+A[i,j+5]
    n3=n3+B[i,j+5]
  }
  k=k+1
  C[1,k]=n0/nsim
  C[2,k]=n1/nsim
  k=k+1
  C[1,k]=n2/nsim
  C[2,k]=n3/nsim
}

#Mostramos la matriz en pantalla con los resultados finales.
#La fila 1 recoge la información relativa a beta0 y
#la fila 2 la información relativa a beta1.
#Las columnas 1 y 2 se refieren al nivel de confianza y amplitud del IC
#resultado de aplicar  mínimos cuadrados.
#Las columnas 3 y 4 se refieren al nivel de confianza y amplitud del IC
#bootstrap-t resultado de aplicar bootstrap a la pareja.
#Las columnas 5 y 6 se refieren al nivel de confianza y amplitud del IC
#tipo percentil resultado de aplicar bootstrap a la pareja.
#Las columnas 7 y 8 se refieren al nivel de confianza y amplitud del IC
#bootstrap-t resultado de aplicar bootstrap a los residuos.
#Las columnas 9 y 10 se refieren al nivel de confianza y amplitud del IC
#tipo percentil resultado de aplicar bootstrap a los residuos.
C
```

A.2. Sección 2.3: Transformaciones del intervalo percentil

```
#Generamos una muestra aleatoria de tamaño 10 distribuida
#según una normal estándar
x=rnorm(10,0,1)

#Valor real del parámetro
exp(0)

#Estimación del parámetro
exp(mean(x))

#Generamos 1000 réplicas bootstrap
nreplicas=1000
library(boot)

#Creamos la función que será el estadístico
bootfun <- function(x,i){
  exp(mean(x[i]))
}

bootE <- boot(x,bootfun,R=nreplicas)

#Mostramos el resumen del modelo de donde obtenemos el error estándar
bootE

#Calculamos el intervalo percentil
boot.ci(bootE, conf = 0.95, type = "perc")

##TRANSFORMACIÓN LOGARITMO##
#Valor real del parámetro
log(exp(0))
#Estimación del parámetro
log(exp(mean(x)))

#Aplicamos bootstrap
bootfun2 <- function(x,i){
  log(exp(mean(x[i])))
}

bootL <- boot(x,bootfun2,R=nreplicas)
bootL

#Calculamos el intervalo percentil
IC=boot.ci(bootL, conf = 0.95, type = "perc")

#Calculamos la exponencial de los extremos del intervalo
exp(IC$perc[4])
exp(IC$perc[5])
```